AD-A246 522

NAVAL POSTGRADUATE SCHOOL
Monterey, California

DTIC
ELECTE
FEB 28 1992
S B D

THESIS

PERFORMANCE ANALYSIS OF IMAGE
MOTION ANALYSIS ALGORITHMS

by

İbrahim Aksu

June, 1991

Thesis Advisor: Jeffrey B. Burl

Approved for public release; distribution is unlimited.

92-04943

92 2 25 179

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY 2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b. OFFICE SYMBOL (if applicable) EC	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000	7b. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (if applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)
PERFORMANCE ANALYSIS OF IMAGE MOTION ANALYSIS ALGORITHMS (U)

12. PERSONAL AUTHOR(S)
AKSU, Ibrahim

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Master's Thesis	FROM TO	1991, JUNE	82

16. SUPPLEMENTARY NOTATION The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the United States Government.

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Extended Kalman Filter; Feature-based; Optical Flow; Accumulative Differencing.

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Computer simulation studies of image motion analysis algorithms are presented. The algorithms are the extended Kalman filter algorithm, linear feature-based algorithms (perspective and orthogonal), and the accumulative differencing algorithm. The simulation studies both using computer generated and real images are conducted to determine the performance of the algorithms on low signal to noise ratio images. Using the results of simulation studies, a comparison of the performance of image motion analysis algorithms is performed.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT [X] UNCLASSIFIED/UNLIMITED [ ] SAME AS RPT. [ ] DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL BURL, Jeffrey B.	22b. TELEPHONE (Include Area Code) (408) 646-2390 22c. OFFICE SYMBOL EC/Bl

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

Approved for public release; distribution is unlimited.

Performance Analysis of Image Motion Analysis Algorithms

by

İbrahim Aksu
Lieutenant Junior Grade, Turkish Navy
B.S., Turkish Naval Academy, 1985

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June, 1991

Author: _____
İbrahim Aksu

Approved by: _____
Jeffrey B. Burl, Thesis Advisor

_____
Roberto Cristi, Second Reader

_____
Michael A. Morgan, Chairman
Department of Electrical and Computer Engineering

ABSTRACT

Computer simulation studies of image motion analysis algorithms are presented. The algorithms are the extended Kalman filter algorithm, linear feature-based algorithms (perspective and orthogonal), and the accumulative differencing algorithm. The simulation studies both using computer generated and real images are conducted to determine the performance of the algorithms on low signal to noise ratio images. Using the results of simulation studies, a comparison of the performance of image motion analysis algorithms is performed.

Accession For
NTIS GRA&I ☑
DTIC TAB ☐
Unannounced ☐
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

TABLE OF CONTENTS

I. INTRODUCTION 1

A. GENERAL 1

B. METHODOLOGIES FOR MOTION ESTIMATION 2

II. EXTENDED KALMAN FILTER ALGORITHM 5

A. GENERAL 5

B. OUTLINE OF THE EKF ALGORITHM 6

C. VELOCITY ESTIMATION 9

D. SIMULATION 14

III. FEATURE-BASED ALGORITHMS 17

A. PROJECTIONS 17

B. MOTION EQUATIONS UNDER PERSPECTIVE PROJECTION 22

C. ERROR ANALYSIS OF THE ALGORITHM 32

D. MOTION EQUATIONS UNDER ORTHOGRAPHIC PROJECTION 43

E. SIMULATION 55

IV. DIFFERENCE IMAGES 57

A. GENERAL 57

B. ACCUMULATIVE DIFFERENCES 60

C. BINARY IMAGES AND VELOCITY ESTIMATION 62

D. SIMULATION 64

V. CONCLUSIONS.....70

LIST OF REFERENCES.....73

INITIAL DISTRUBUTION LIST.....75

I. INTRODUCTION

A. GENERAL

Determining the relative motion between an observer and his environment is a major problem in computer vision. Its applications include mobile robot navigation and the monitoring of dynamic processes. Motion estimation also has many applications in image processing, such as coding, restoration, and reducing the noise by temporal filtering.

In computer vision, motion in images is recovered from a time ordered sequence of images. The relative motion between the objects in a scene and a camera, gives rise to the apparent motion of the objects in a sequence of images. This motion may be characterized by observing the apparent motion of a discrete set of features or brightness patterns in the images. The objective of motion analysis is the derivation of the motion of the objects in the scene through the analysis of motion features or brightness patterns associated with objects in the sequence of images.

A closely related subject to motion estimation is the *estimation of structure* of the imaged scene. Although structure may be computed independent of motion via stereo vision, knowledge of the object motion can facilitate establishment of feature correspondences within a pair of stereo images, thus aiding the determination of structure. Indeed, psychological researchers have shown that apparent motion is a clue used by the human visual system for computing the scene structure [Ref. 1]. This close relationship between the estimation of structure and the estimation of motion has combined these two problems.

In the following sections, the fundamental principles of several approaches for the estimation of motion and structure will be discussed.

B. METHODOLOGIES FOR MOTION ESTIMATION

Two distinct approaches have been developed for the computation of motion:

1. Methods Based on the Use of Image Position

This method is based on extracting a set of relatively sparse, but highly discriminatory, two-dimensional features in the images corresponding to three-dimensional object features, such as corners or occluding boundaries of surfaces. Such points, lines and/or curves are extracted from each frame and then inter-frame correspondence is established between these features. The observed displacement of the 2D image features are used to solve the resulting motion equations. Constraints are formulated based on a rigid body motion assumption. This method assumes that correspondence is available between features extracted from one image in a sequence of images and those extracted from the next image. Establishing and maintaining such correspondence is a very hard problem. The ambiguity is produced by the effects of occlusion and noise which cause features to appear or disappear and also give rise to false features. Some of the techniques developed for solving the corresponding problem for stereo vision and optical flow algorithms may be applied to this problem [Ref. 1].

Analysis algorithms that rely on feature correspondence are termed feature-based algorithms. The feature-based algorithms will be explained in Chapter III in more detail. First projection methods, then the motion analysis algorithms [Ref. 2], [Ref. 3]

based on these projection methods will be explained. An analysis of motion parameter error caused by correspondence errors will also be presented in chapter III.

2. Methods Based On Optical Flow

Optical flow is the two-dimensional field of instantaneous velocities of brightness values (gray levels) in the image plane. The optical flow techniques mainly rely on local spatial and temporal derivatives of image brightness values. There are several methods which have been proposed for computing optical flow of time-varying images. These methods are mainly based on:

a. Local features

The first step of this method is the detection of local features in each image. After obtaining the features a pair-wise matching between corresponding features in two frames must be obtained that minimizes an appropriate cost function.

b. Spatitemporal gradient

This method uses the first order spatial and temporal differentials of time varying images to estimate at each image point the component of motion in the direction of maximally increasing gray-scale intensity. This method is based on the equation:

$$\frac{\delta f}{\delta x}u+\frac{\delta f}{\delta y}v+\frac{\delta f}{\delta t}=0, \tag{1.1}$$

where f is the image function, t is time, δ is the partial derivative operator, u and v are the x and y components of the optical velocity [Ref. 4].

Equation 1.1 alone is not sufficient to determine the optical velocities. Some constraints must be imposed, for example:

1. Optical flow is smooth.
2. Optical flow is constant and continuous over entire segments in image.
3. Motion is restricted (e.g., *planar motion*).

Using these constraints, some solutions are proposed [Ref. 5], [Ref. 6].

c. Fourier Phase Approach

This method uses the shift property of the Fourier transform, and is the most appropriate method for determining the motion of a single object moving across a uniform background [Ref. 4]. This method is restrictive because only a single non-localized velocity vector is obtained for each image frame.

The computation of optical flow requires the evaluation of first and second partial derivatives of image brightness values and also of the optical flow. The real images are noisy, in general. The evaluation of derivatives is a noise enhancing operation. As we increase the order of the derivative, we increase the sensitivity of noise. Also because of occlusion there may be some discontinuities in the optical flow, these regions must be detected reliably, otherwise the continuity assumption is violated.

A new Fourier phase approach proposed by Burl [Ref. 7] based on the extended Kalman filter is discussed in Chapter II. The extended Kalman filter is implemented in spatial frequency domain and provides an estimate of the object velocity. Accumulative difference and binary image approaches are discussed in Chapter IV. Chapter V compares the performance of the algorithms for low signal to noise ratio (SNR) images. A discussion about the results of the computer simulations will also be presented at that chapter.

II. EXTENDED KALMAN FILTER ALGORITHM

A. GENERAL

An extended Kalman filter (EKF) is used to estimate the velocity of an object moving across an image frame and to reduce the effect of noise. The algorithm is proposed by Burl [Ref. 7] and the image sequences are modeled by dynamic nonlinear state equations. A simple dynamic model consists of a shift operator with the shift given by the velocity times the sample time. This model is given in both the spatial and spatial frequency domains. Application of EKF in the spatial domain requires a prohibitive number of calculations but implementation of the EKF in the spatial frequency domain reduces the number of calculations by a great amount. The resulting algorithm, referred to as the parallel extended Kalman filter (PEKF), is developed in detail and a number of practical considerations are presented in the Reference 7. As the velocity estimation error approaches zero, the EKF is shown to converge to a parallel set of third-order extended Kalman filters. This structure is referred to as the modified extended Kalman filter (MEKF). The single frequency EKFs are combined to yield an estimate of the velocity of the moving object using weighted least square algorithm. An ambiguity of multiplies of 2π in the frequency-velocity estimates is addressed in Reference 7. A proposed a two-step algorithm to solve the ambiguity utilizing some structures of the problem is also given in Reference 7. First, the object velocity is estimated using the subset of the spatial frequencies where there is no ambiguity, then this velocity estimation is used to estimate the ambiguities for other spatial frequencies. Another approaches to solve the ambiguity problem will be presented in Section II.C using the properties of

EKF and the problem structure. An outline of the EKF algorithm is given in Section II.B. The simulation results of the proposed algorithm with varying SNR images is given in Section II.D.

B. OUTLINE OF THE EKF ALGORITHM

In the spatial domain, successive image frames composed of N by N pixels and containing a moving object is modeled by a nonlinear state space equation:

$$x(k+1)=\begin{bmatrix} f(k+1) \\ v(k+1) \end{bmatrix}=\begin{bmatrix} S(v) & 0 \\ 0 & I \end{bmatrix}\begin{bmatrix} f(k) \\ v(k) \end{bmatrix}+\begin{bmatrix} n_f(k) \\ n_v(k) \end{bmatrix}, \tag{2.1}$$

where $x(k)$ is the state of the system, $f(k) \in R^{(NN)}$ is composed of amplitudes of each pixel at time k stacked into a vector, $v \in R^2$ is the velocity vector in pixels per sampling time, and $S(v)$ is a two-dimensional shift operator with the magnitude and direction being a function of the velocity vector, and $n_f(k)$ and $n_v(k)$ are the image noise and the velocity noise, respectively.

The corresponding measurement equation is:

$$y(k)=f(k)+\nu(k)=[I \;\; 0]x(k)+\nu(k), \tag{2.2}$$

where $y \in R^{(NN)}$ is the measured image and $\nu \in R^{(NN)}$ is a zero mean, white, Gaussian random process with a covariance matrix $R_{\nu\nu}=\sigma^2 I$.

The spatial domain model described by Equations 2.1 and 2.2 is transformed to the spatial frequency domain by using the two-dimensional discrete Fourier transform. The state equation in the spatial frequency domain is then defined:

$$\begin{bmatrix} F(k+1) \\ V(k+1) \end{bmatrix} = \begin{bmatrix} D(v) & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} F(k) \\ V(k) \end{bmatrix} + \begin{bmatrix} Z_F(k) \\ Z_V(k) \end{bmatrix}, \tag{2.3}$$

where $F(k)$ is the fourier transform of $f(k)$, $D(v)$ is the transformed shift operator for a specific spatial frequency, and $Z_F(k)$ is the transformed image plant noise, $Z_V(k)$ is the transformed velocity noise.

The corresponding measurement equation then becomes:

$$Y(k) = F(k) + N(k) = [I \;\; 0]X(k) + N(k), \tag{2.4}$$

where $Y(k)$ and $N(k)$ are the Fourier transform of $y(k)$ and $\nu(k)$, respectively.

The EKF uses the spatial frequency domain state equations, Equations 2.3 and 2.4. The linearized state equation is given as,

$$\delta X(k+1) = \mathrm{A}(X_0)\delta X(k) + Z(k), \tag{2.5}$$

where X_0 is the current estimate and $\delta X(k) = X(k) - X_0$, $A(X_0)$ is the Jacobian of the nonlinear state equations about X_0.

A summary of EKF structure is given below from References 8 and 9:

State estimation:

$$\hat{X}(k+1|k) = a(\hat{X}(k|k), 0); \tag{2.6}$$

Covariance estimation:

$$P(k+1|k) = \mathrm{A}(\hat{X}(k|k))P(k|k)\mathrm{A}^T(\hat{X}(k|k)) + Q_{ZZ}; \tag{2.7}$$

Innovation:

$$e(k|k) = Y(k+1) - \hat{Y}(k+1|k) = Y(k+1) - C\hat{X}(k+1|k); \tag{2.8}$$

Kalman Gain:

$$G(k+1) = P(k+1|k)C^T\left[C\ P(k+1|k)\ C^T + R_{NN}\right]^{-1}; \tag{2.9}$$

State correction:

$$\hat{X}(k+1|k+1) = \hat{X}(k+1|k) + G(k+1)[Y(k+1) - \hat{Y}(k+1|k)]; \tag{2.10}$$

Covariance correction:

$$P(k+1|k+1) = [\ I - G(k+1)C]P(k+1|k). \tag{2.11}$$

The parallel extended Kalman filter structure is given in Figure 2.1. Reference 7 outlines how the EKF converges to the MEKF as the velocity estimation error approaches zero.

The state vector for a specific spatial frequency is defined:

$$X_i(k) = \begin{bmatrix} X_{i1}(k) \\ X_{i2}(k) \\ X_{i3}(k) \end{bmatrix} = \begin{bmatrix} Re(F_i(k)) \\ Im(F_i(k) \\ \omega_i^T v \end{bmatrix}, \tag{2.12}$$

where $X_{i1}(k)$ and $X_{i2}(k)$ are the Fourier coefficients of a specific spatial frequency and X_{i3} is the frequency-velocity product.

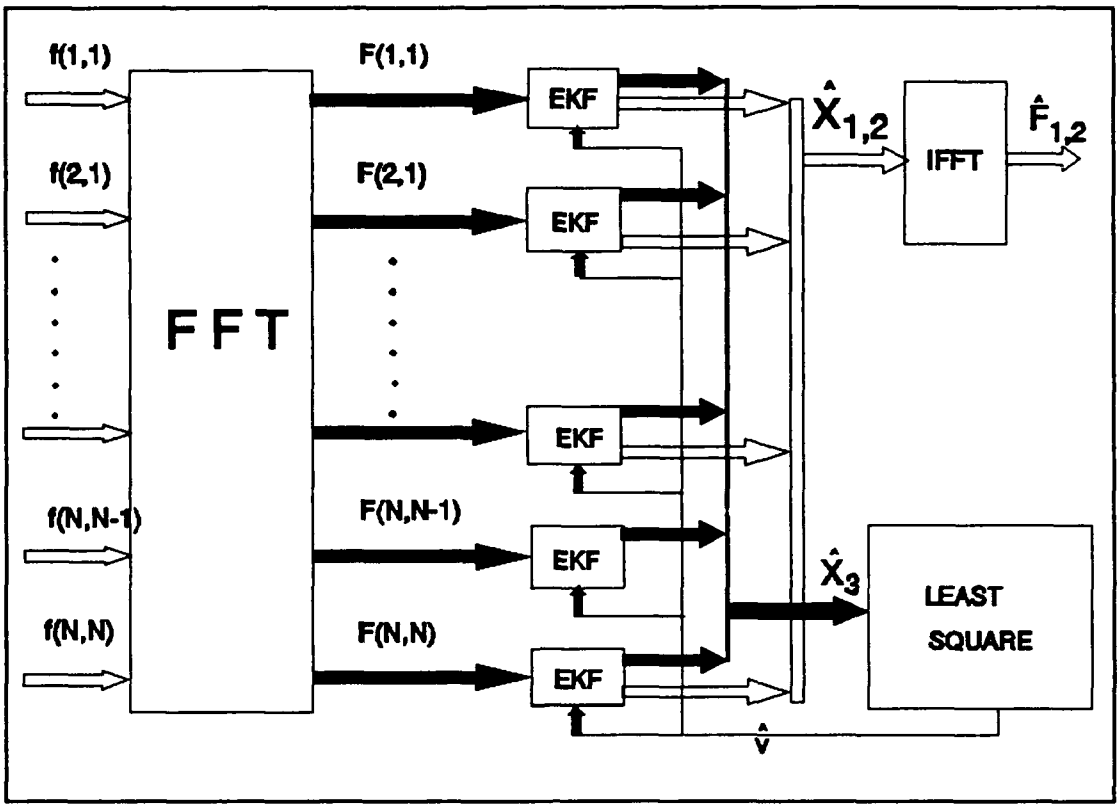

Figure 2.1 The Modified Extended Kalman Filter.

C. VELOCITY ESTIMATION

1. Weighted Least Square

In Equation 2.12, X_{i3} is the estimated frequency-velocity product. These estimates can be combined to obtain the estimate of the velocity of the moving object. This estimation algorithm is complicated by an ambiguity of multiples of 2π. The frequency-velocity product estimates can be modelled as,

$$\hat{X}_3(k|k)=\Omega v+w+2\pi m, \tag{2.13}$$

where $X_3(k \mid k)$ is the estimated frequency-velocity product, Ω contains the spatial frequencies, and w is a zero mean, random vector, with the estimated covariance:

$$\Sigma = E[ww^T] = diag[P_{133}, P_{233}, \ldots, P_{N^2 33}]. \tag{2.14}$$

The subscript $i33$ denotes the [3,3] component of the state estimation error covariance matrix of the ith single frequency EKF. The vector m models the ambiguity whose elements are integer and constrained to be:

$$|m_i| \leq m_{max} = N, \tag{2.15}$$

since the object is constrained to remain within the image.

The estimated $X_3(k \mid k)$ can be used to estimate the velocity that minimizes the weighted sum of the square error:

$$J = (\hat{X}_3(k|k) - \Omega v - 2\pi m)^T \Sigma^{-1} (\hat{X}_3(k|k) - \Omega v - 2\pi m). \tag{2.16}$$

The solution for v can be found by setting the gradient of J with respect to v to zero:

$$\frac{\partial J}{\partial v} = \Omega^T \Sigma^{-1} \Omega v + \Omega^T \Sigma^{-1} (2\pi m) - \Omega^T \Sigma^{-1} \hat{X}_3(k|k) = 0. \tag{2.17}$$

From Equation 2.17:

$$v = (\Omega^T \Sigma^{-1} \Omega)^{-1} (\Omega^T \Sigma^{-1} \hat{X}_3(k|k) - \Omega^T \Sigma^{-1} (2\pi m)). \tag{2.18}$$

The search for optimal vector m can be simplified by utilizing some conditions of the problem as explained in Reference 7. From these conditions it is found that the m_is are equal to zero for almost all spatial frequencies with magnitude given:

$$\|\omega\| < \frac{2\pi N}{\|v_{max}\|}. \tag{2.19}$$

This subset of spatial frequencies can be used to estimate the unambiguous velocity. Then this velocity can be used to estimate the m_is for spatial frequencies that do not meet the condition in Equation 2.19. The solution for vector m can be found by equating the gradient of J in Equation 2.16 with respect to m to zero:

$$\frac{\partial J}{\partial m} = 2\pi m + \Omega v - \hat{X}_3(k|k) = 0. \tag{2.20}$$

Using Equation 2.19:

$$\hat{m}_i = Round\left(\frac{1}{2\pi}\left[\hat{X}_3(k|k) - \omega_i^T \hat{v}\right]\right), \tag{2.21}$$

where *Round(·)* equals to nearest integer. Then Equation 2.18 can be used to estimate the v vector that minimizes the Equation 2.16. The frequency-velocity product estimates are fed back to single frequency EKFs. Once the filter has converged, the values of m will all be equal to zero.

2. Constrained Least Square

Another approach to estimating the object velocity vector from Equation 2.13 is the constrained least squares method [Ref. 10]. We can select one of the frequency component satisfying Equation 2.19 denoted by C, and the corresponding estimated frequency-velocity product denoted by d, as a constraint equation. Then the linear constraint equation becomes:

$$Cv = d. \tag{2.22}$$

The problem now becomes:

$$\min_{v}(\hat{X}_3(k|k)-\Omega v)^T(\hat{X}_3(k|k)-\Omega v) \tag{2.23}$$

$$\text{Subject to} \quad Cv = d.$$

From Equation 2.23 we can form the following Lagrangian J:

$$J=\frac{1}{2}(\hat{X}_3(k|k)-\Omega v)^T(\hat{X}_3(k|k)-\Omega v)-\lambda(Cv-d). \tag{2.24}$$

The solution for v can be found by equating the gradient of J with respect to v to zero:

$$\frac{\partial J}{\partial v}=\Omega^T\Omega v-\Omega^T\hat{X}_3(k|k)-\lambda C^T=0. \tag{2.25}$$

The solution for v then becomes:

$$v=(\Omega^T\Omega)^{-1}\Omega^T\hat{X}_3(k|k)+(\Omega^T\Omega)^{-1}C^T\lambda. \tag{2.26}$$

We can write this equation as,

$$v_{CLS}=v_{LS}+(\Omega^T\Omega)^{-1}C^T\lambda, \tag{2.27}$$

where v_{CLS} denotes constraint least square solution, and v_{LS} denotes the least square solution. We can solve for λ by invoking the constraint defined in Equation 2.22:

$$Cv_{CLS}=Cv_{LS}+C(\Omega^T\Omega)^{-1}C^T\lambda=d. \tag{2.28}$$

From Equation 2.28:

$$\lambda = \left[C(\Omega^T\Omega)^{-1}C^T\right]^{-1}(d - Cv_{LS}). \tag{2.29}$$

Substituting Equation 2.29 into 2.27, we obtain the constraint least square solution:

$$v_{CLS} = v_{LS} + (\Omega^T\Omega)^{-1}C^T\left[C(\Omega^T\Omega)^{-1}C^T\right]^{-1}(d - Cv_{LS}). \tag{2.30}$$

3. Adding Fictitious Noise

Since the ambiguity is not modeled in the single frequency EKFs, the system may be assumed to have a modeling problem. This problem may cause the true estimation errors to become infinite if we don't process the frequency-velocity estimates by one of the methods explained previously. Another way of solving this problem is to add fictitious process noise to the system as explained in References 8 and 9.

The true frequency-velocity estimations are:

$$X_{3_{TRUE}} = \Omega v_{TRUE}. \tag{2.31}$$

We can add $\Omega\mu$, where μ is a 2 by 1 constant vector, to the frequency-velocity estimates at every step. The velocity estimate then consists of the true velocity plus a bias. This bias is eliminated easily after the system has converged by using the fact that velocity of the object is multiple of integers since the velocity is assumed to be pixel/frame. If the elements of vector μ is positive and the object velocity is negative then the algorithm will estimate the object velocity as true velocity plus N, where N is the size of image since the object motion is assumed to be periodic. Using that method eliminates the maximum velocity constraint which is imposed from Equation 2.15.

D. SIMULATION

The velocity estimation algorithms are simulated using varying signal to noise ratio (SNR) images. A computer generated square object has been moved on a 16 by 16 image. Artificial noise is added to the image for each experiment. The SNR of each image is calculated as signal power over noise power in dB units. Relative translation vector error is obtained as the norm of error over the norm of true vector for each iteration except the first one, since it is assumed the initial velocity estimate is zero. Figure 2.2 shows the result of experiments that have been performed using 6 iterations for each SNR and using a positive velocity and a positive μ vector for the fictitious noise algorithm. The two-step weighted least square algorithm needs more iterations than 6 to converge to the true velocity for low SNR images [Ref. 11]. The relative errors of this algorithm for low SNR are larger but they are smaller for high SNR images. The adding fictitious noise algorithm converges to the true velocity faster, since the added fictitious noise is in the shape of the expected value of true frequency-velocity product and the relative error becomes smaller for this algorithm.

A second experiment is performed to indicate how the EKF algorithm enhances the image quality. The SNR of the input image and the SNR of the image estimated (output) by EKFs are compared. Figure 2.3 shows that for SNRs less than 10 dB there is an improvement in the SNR of the output image and this improvement is constant and about 7 dB for very low SNR images.

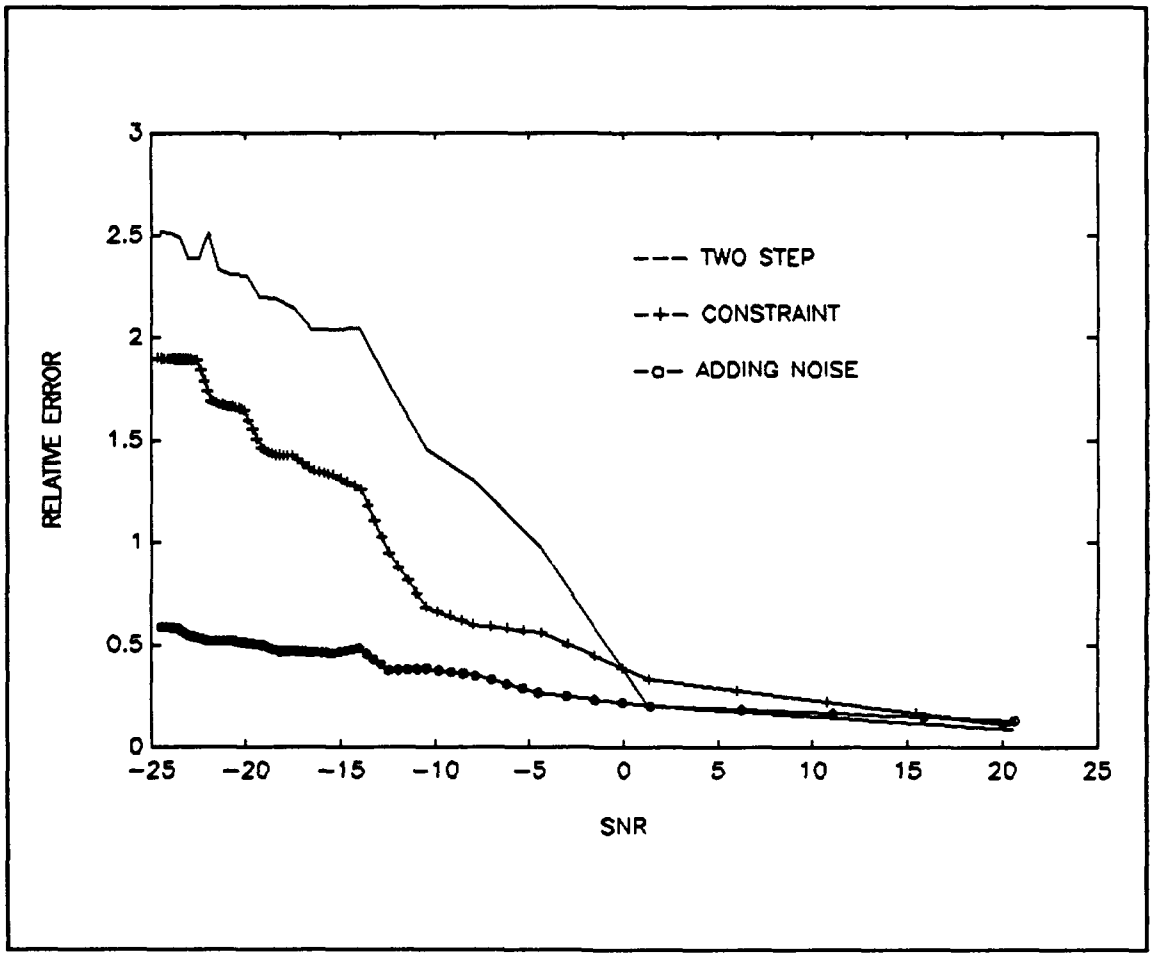

Figure 2.2 Comparison of velocity estimation methods.

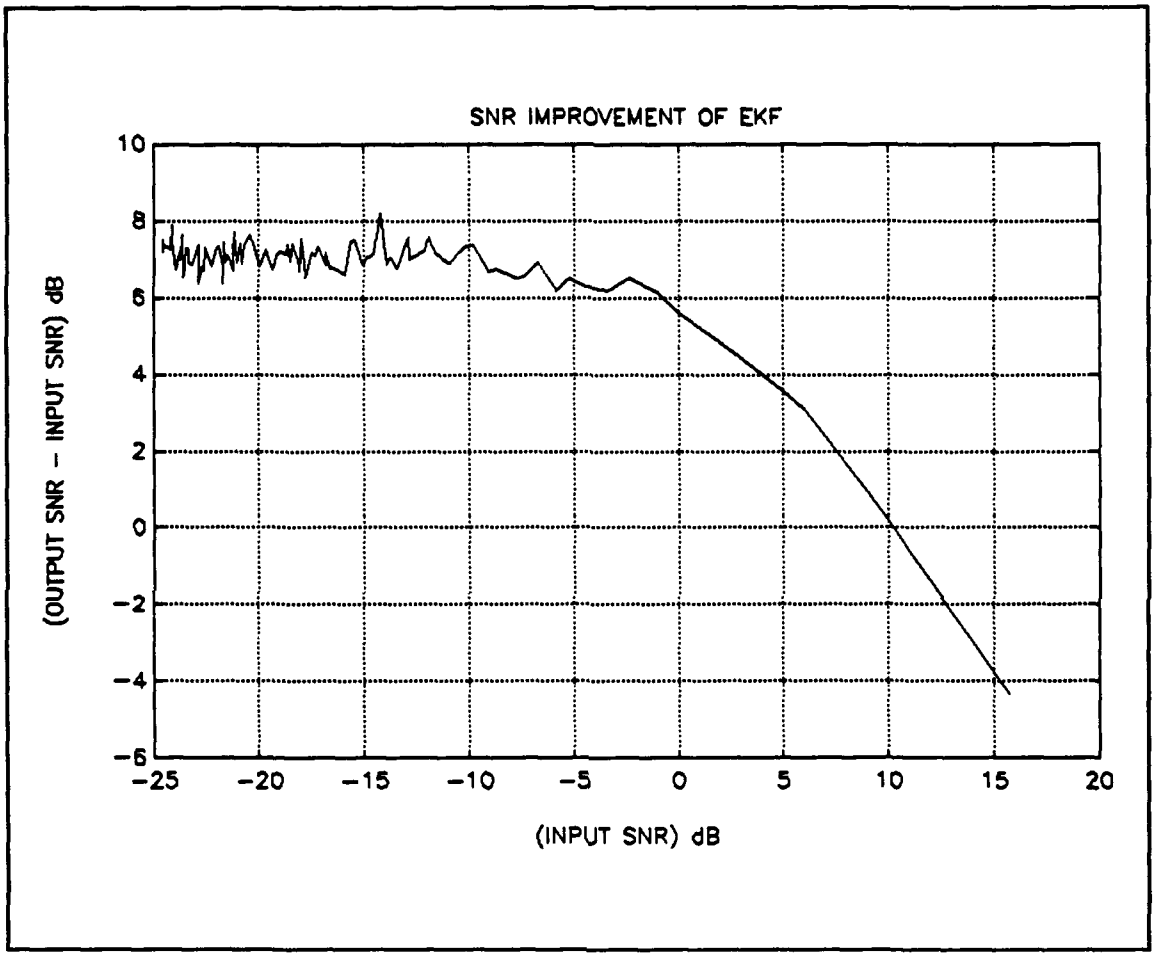

Figure 2.3 SNR improvement of EKF algorithm.

III. FEATURE-BASED ALGORITHMS

A. PROJECTIONS

1. General

Image analysis requires an understanding of how the image was formed. How a two-dimensional pattern of brightness is produced in an optical image-forming system can be studied in two parts: first, we need to find the geometric correspondence between points in the scene and points in the image; then we must figure out what determines the brightness at a particular point in the image. The geometric correspondences are termed projections.

Projections transform points in a coordinate system of dimension n into points in a coordinate system of dimension less than n [Ref. 12]. Here, we will use the projection from 3D to 2D. The projection of a 3D object is defined by straight projection rays (called *projectors*) emanating from a center of projection, passing through each point of the object, and intersecting a projection plane to form the projection.

Projections can be divided into two basic classes: *perspective* and *parallel.* The distinction is in the relation of the center of projection to the projection plane. If the distance from the one to the other is finite, then the projection is perspective. If the distance is infinite, the projection is parallel. Figure 3.1 shows these two cases. The parallel projection is so named because, with the center of projection infinitely distant, the projectors are parallel. When defining a perspective projection, we explicitly specify its center of projection; for a parallel projection, we give its direction of projection.

The visual effect of a perspective projection is similar to that of photographic systems and of the human visual system. The size of the perspective projection of an object varies inversely with the distance of that object from the center of the projection plane.

The parallel projection is a less realistic view because perspective

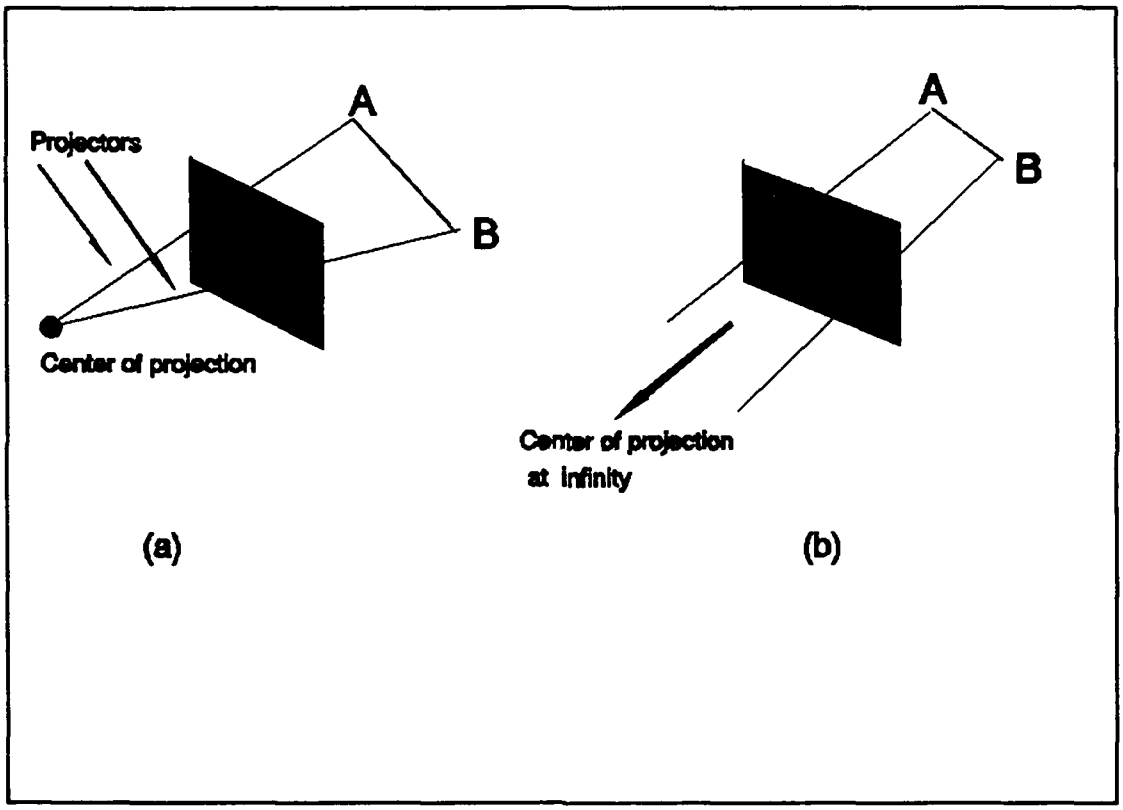

Figure 3.1 (a) Line AB and its perspective projection. (b) Line AB and its parallel projection.

foreshortening is lacking. The advantages of parallel projection is that the projection can be used for exact measurements and parallel lines remain parallel.

2. Perspective Projections

We can approximate an optical imaging system with an ideal pinhole at a fixed distance in front of an image plane. Assume that only light coming through the pinhole can reach the image plane. Since light travels along straight lines, each point in the image corresponds to a particular direction defined by a ray from that point through the pinhole (Figure 3.2).

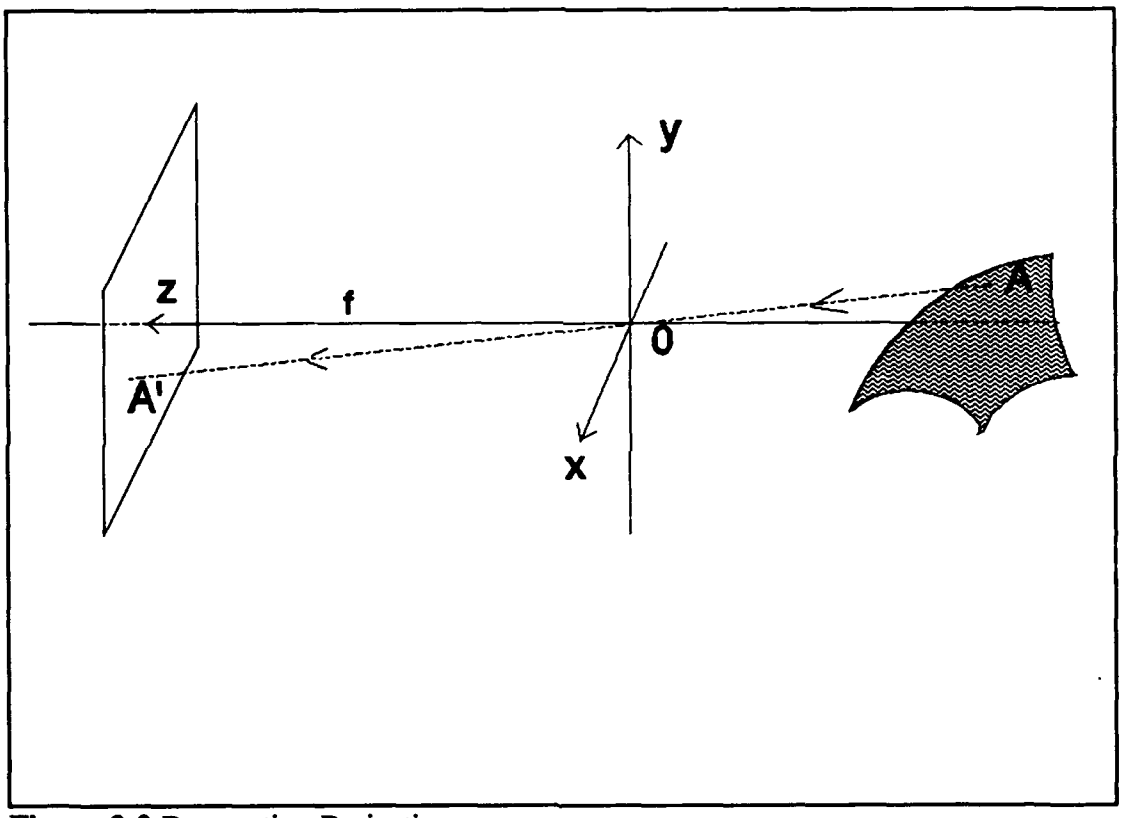

Figure 3.2 Perspective Projection

We define the *optical axis* in this system to be perpendicular to the line from the pinhole to the image plane. We can introduce a cartesian coordinate system with the origin at the projection point and the z-axis aligned with the optical axis and pointing toward the image. Let $V = (X,Y,Z)$, the vector connecting O to A and $V' = (x,y,f)$, the

vector connecting O to A', with f the focal length, that is, the distance of the image plane from nodal point O; and (x,y) are the coordinates of the point A' on the image plane in the coordinate system with origin at the point of the intersection of the image plane with the optical axis, and axes x and y parallel to the axis of the camera coordinate system OX and OY. By similar triangles, we can easily write:

$$x=\frac{fX}{Z}, y=\frac{fY}{Z}. \tag{3.1}$$

These equations relate the image coordinates to the world coordinates of a point. Further, to simplify the equations we may assume $f = 1$ without loss of generality.

3. Parallel Projections

Parallel projections are categorized into two types depending on the relation between the direction of projection and the normal to the projection plane. In *orthographic* parallel projections (Figure 3.3), these directions are the same, so the direction of the projection is normal to the projection plane. In *oblique* parallel projections, the projection plane is normal but the direction of the projection is different. If the direction of projection makes a 45° angle with the projection plane, it is called *cavalier*, if it makes a 63.4° angle, it is called *cabinet* projection. For our case orthographic projection is considered.

We have a plane that lies parallel to the image plane at $Z = Z_0$ in the previous perspective projection model. We define the magnification m, as a ratio of the distance between two points measured in the image to the distance between the

corresponding points on the plane. Consider a small interval on the plane $(dX,dY,0)^T$ and the corresponding small interval $(dx,dy,0)^T$ in the image. Then:

$$m=\frac{\sqrt{dx^2+dy^2}}{\sqrt{dX^2+dY^2}}=\frac{f}{-Z_0} \tag{3.2}$$

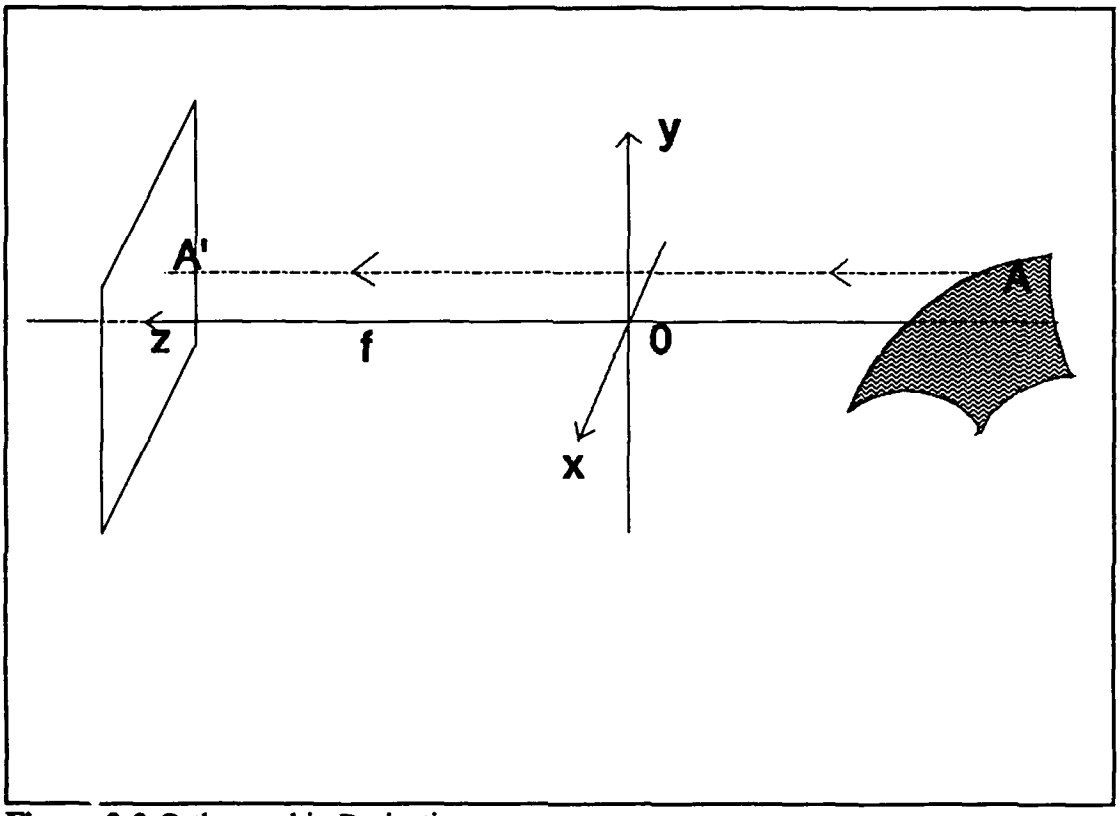

Figure 3.3 Orthographic Projection

where $-Z_0$ is the distance of the plane from the pinhole. The magnification is the same for all points in the plane.

A small object at an average distance $-Z_0$ will produce an image that is magnified by m. The magnification is approximately constant when the depth range of

the scene is small relative to the average distance of the surfaces from the camera. In this case we can simply write for projection equations, that

$$x=mX, y=mY. \tag{3.3}$$

with $m = f/(-Z_0)$ and $-Z_0$ is the average value of the depth $-Z$. Often the scaling factor m is set to 1 or -1 for convenience. Then we can further simplify the equations to become:

$$x=X,\ y=Y. \tag{3.4}$$

These equations model the orthographic projection model (Figure 3.3), where the rays are parallel to the optical axis or the focal length is infinite.

The difference between perspective and orthographic projection is small when the distance to the scene is much larger than the variation in distance among objects in the scene.

B. MOTION EQUATIONS UNDER PERSPECTIVE PROJECTION

We can analyze the relation between the image plane motion and the corresponding three-dimensional motion for the case of perspective projection. For simplicity f (focal length) is assumed to be unity and image plane is assumed to be stationary.

A rigid body with coordinates $(X,Y,Z)^T$ moves with a translational velocity $V_T = (U,V,W)^T$ and rotational velocity $\Omega = (A,B,C)^T$. From kinematics the three-dimensional velocity of any point on the surface can be written as:

$$V=(\frac{dX}{dt},\frac{dY}{dt},\frac{dZ}{dt})=V_T+\Omega x(X,Y,Z). \tag{3.5}$$

We can write this equation in component form as:

$$\dot{X}=U+BZ-CY; \tag{3.6}$$

$$\dot{Y}=V+CZ-AZ; \tag{3.7}$$

$$\dot{Z}=W+AY-BX. \tag{3.8}$$

Let (x,y) denote the coordinates of a point in the image plane. As explained before for perspective projection the relation between object point P, and the corresponding image point p is:

$$x=\frac{X}{Z}; \tag{3.9}$$

$$y=\frac{Y}{Z}. \tag{3.10}$$

The *optical flow* at each point in the image plane is the instantaneous velocity of the brightness pattern at that point. So the optical flow of the point (x,y) can be denoted by (u,v) where:

$$u=\dot{x}, \quad v=\dot{y}. \tag{3.11}$$

Differentiating Equations 3.9 and 3.10 with respect to time and using Equations 3.6, 3.7 and 3.8 we obtain:

$$u=\frac{dx}{dt}=\frac{\dot{X}}{Z}-\frac{X\dot{Z}}{Z^2}=\frac{U}{Z}+B-\frac{CY}{Z}-\frac{X}{Z^2}(W+AY-BX)$$
$$=\frac{U-xW}{Z}+B(1+x^2)-Axy-Cy. \tag{3.12}$$

In the same way:

$$v=\frac{V-yW}{Z}+Bxy-A(y^2+1)+Cx. \tag{3.13}$$

We can also write these equations in the form of:

$$u=u_t+u_r, \quad v=v_t+v_r \tag{3.14}$$

where (u_t, v_t) denotes translational component of the optical flow and (u_r, v_r) the rotational component.

$$u_t=\frac{U-xW}{Z}, \quad u_r=B(1+x^2)-Axy-Cy; \tag{3.15a}$$

$$v_t=\frac{V-yW}{Z}, \quad v_r=Bxy-A(y^2+1)+Cx. \tag{3.15b}$$

From Equations 3.12 and 3.13 we can eliminate the unknown depth variable Z. From Equation 3.12:

$$Zu=U-xW+B(1+x^2)Z-AxyZ-CyZ \Rightarrow Z=\frac{U-xW}{u-B(1+x^2)+Axy+Cy}, \tag{3.16}$$

and from Equation 3.13:

$$Z=\frac{V-yW}{v+A(y^2+1)-Bxy-Cx}. \tag{3.17}$$

From these two:

$$\frac{U-xW}{V-yW}=\frac{u-B(1+x^2)+Axy+Cy}{v+A(y^2+1)-Bxy-Cx}. \tag{3.18}$$

Equation 3.18 describes the constraint imposed by the measured value of the optical flow (u,v) at any image point (x,y) on the six motion parameters (U,V,W,A,B,C).

Consider one point $P = (X,Y,Z)$ before the motion with image point (x,y). Suppose that the point moves with a general motion, and goes to point $P' = (X',Y',Z')$ with image point (x',y'). Any three-dimensional rigid body motion is equivalent to a rotation by an angle Θ around an axis through the origin with directional cosines n_1, n_2, n_3 followed by a translation $T=(\Delta X,\Delta Y,\Delta Z)^T$. The relation between coordinates of the point before and after the transformation is given by:

$$(X',Y',Z')^T = R(X,Y,Z)^T+T, \tag{3.19}$$

where R is 3 by 3 orthonormal matrix of first kind *(i.e., det(R)=1)*, and it is defined as [Ref. 13],

$$R=\begin{bmatrix} n_1^2+(1-n_1^2)\cos\theta & n_1n_2(1-\cos\theta)+n_3\sin\theta & n_1n_3(1-\cos\theta)-n_2\sin\theta \\ n_1n_2(1-\cos\theta)-n_3\sin\theta & n_2^2+(1-n_2^2)\cos\theta & n_2n_3(1-\cos\theta)+n_1\sin\theta \\ n_1n_3(1-\cos\theta)+n_2\sin\theta & n_2n_3(1-\cos\theta)-n_1\sin\theta & n_3^2+(1-n_3^2)\cos\theta \end{bmatrix}. \tag{3.20}$$

For image vectors the transformation becomes:

$$Z'(x',y',1)^T = ZR(x,y,1)^T + T. \tag{3.21}$$

If $\|T\| \neq 0$, where $\|\bullet\|$ denotes Euiclidean norm, Equation 3.21 can be written:

$$\begin{aligned} \frac{Z'}{\|T\|}(x',y',1)^T &= R\frac{Z}{\|T\|}(x,y,1)^T + \hat{T}, \\ \textit{where} \quad \hat{T} &= \frac{T}{\|T\|}. \end{aligned} \tag{3.22}$$

We can establish a general relationship between the two sets of image coordinates at this point. Using this relationship we can then proceed to solve the motion equations and obtain the structure of the scene.

From Equation 3.19 we can write:

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} \tag{3.23}$$

and in component form:

$$X' = r_{11}X + r_{12}Y + r_{13}Z + \Delta X; \tag{3.24a}$$

$$Y' = r_{21}X + r_{22}Y + r_{23}Z + \Delta Y; \tag{3.24b}$$

$$Z' = r_{31}X + r_{32}Y + r_{33}Z + \Delta Z. \tag{3.24c}$$

From the perspective projection property, Equations 3.9 and 3.10:

$$x'=\frac{X'}{Z'}=\frac{r_{11}X+r_{12}Y+r_{13}Z+\Delta X}{r_{31}X+r_{32}Y+r_{33}Z}=\frac{(r_{11}x+r_{12}y+r_{13})Z+\Delta X}{(r_{31}x+r_{32}y+r_{33})Z+\Delta Z}; \tag{3.25}$$

As we did before we can eliminate the depth variable Z from these two equations.

$$y'=\frac{Y'}{Z'}=\frac{r_{21}X+r_{22}Y+r_{23}Z+\Delta Y}{r_{31}X+r_{32}Y+r_{33}Z+\Delta Z}=\frac{(r_{21}x+r_{22}y+r_{23})Z+\Delta Y}{(r_{31}x+r_{32}y+r_{33})Z+\Delta Z}. \tag{3.26}$$

From Equation 3.25:

$$Z=\frac{\Delta X-x'\Delta Z}{x'(r_{31}x+r_{32}y+r_{33})-(r_{11}x+r_{12}y+r_{13})}. \tag{3.27}$$

From Equation 3.26:

$$Z=\frac{\Delta Y-y'\Delta Z}{y'(r_{31}x+r_{32}y+r_{33})-(r_{21}x+r_{22}y+r_{23})}. \tag{3.28}$$

Equating these two equation and solving, we obtain:

$$\begin{aligned}
&xx'(\Delta Zr_{21}-\Delta Yr_{31})+xy'(\Delta Xr_{31}-\Delta Zr_{11})+x(\Delta Yr_{11}-\Delta Xr_{21})+\\
&yx'(\Delta Zr_{22}-\Delta Yr_{32})+yy'(\Delta Xr_{32}-\Delta Zr_{12})+y(\Delta Yr_{12}-\Delta Xr_{22})+\\
&x'(\Delta Zr_{23}-\Delta Yr_{33})+y'(\Delta Xr_{33}-\Delta Zr_{13})+(\Delta Yr_{13}-\Delta Xr_{23})=0.
\end{aligned} \tag{3.29}$$

Also from Equation 3.29, it is possible to show this relation as,

$$[x' \;\; y' \;\; 1]E\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}=0, \tag{3.30}$$

where

$$E=\begin{bmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix}, \tag{3.30a}$$

with

$$\begin{aligned} &e_1=\Delta Zr_{21}-\Delta Yr_{31},\; e_2=\Delta Zr_{22}-\Delta Yr_{32},\; e_3=\Delta Zr_{23}-\Delta Yr_{33}; \\ &e_4=\Delta Xr_{31}-\Delta Zr_{11},\; e_5=\Delta Xr_{32}-\Delta Zr_{21},\; e_6=\Delta Xr_{33}-\Delta Zr_{13}; \\ &e_7=\Delta Yr_{11}-\Delta Xr_{31},\; e_8=\Delta Yr_{21}-\Delta Xr_{22},\; e_9=\Delta Yr_{31}-\Delta Xr_{23}. \end{aligned} \tag{3.31}$$

The elements of E are called *essential parameters* defined in terms of motion parameters [Ref. 14]. Equation 3.31 relates image coordinates of feature points and the elements of the matrix E. Since these equations are linear and homogeneous in ΔX, ΔY, and ΔZ, the essential parameters can be determined up to a scale factor and the sign of the matrix E is arbitrary. We can solve for motion parameters from these essential parameters. The relative depth (depth scaled by magnitude of translation) of each point can then be determined from the motion parameters and the observed projections of the points. Note that if we assume motion is only translational then the R matrix will be the identity and the matrix E will become:

$$E_t = \begin{bmatrix} 0 & -\Delta Z & \Delta Y \\ \Delta Z & 0 & -\Delta X \\ -\Delta Y & \Delta X & 0 \end{bmatrix} \tag{3.32}$$

which is a *skew-symmetric* matrix.

In general, it is possible to represent the matrix E in the form of:

$$E = E_t R = [E_t x R_1 \;\; E_t x R_2 \;\; E_t x R_3] = [E_1 \;\; E_2 \;\; E_3], \tag{3.33}$$

where $R=[R_1 \;\; R_2 \;\; R_3]$. From Equation 3.22 it can be seen that the magnitude of translational vector ($\|T\|$) and absolute depths of the object points (Z and Z') can not be determined from monocular vision. Equation 3.22 will still hold when $\|T\|$, Z and Z' are multiplied by any positive constant. Since only the direction of T is known and E can be defined up to a scale factor we can normalize things by assuming $\|T\| = 1$. This implies $\|E\|^2 = 2$ which can be proved:

$$\begin{aligned} \|E\|^2 &= trace\{EE^T\} = trace\{E_t R (E_t R)^T\} \\ &= trace\{E_t R R^T E_t^T\} = trace\{E_t E_t^T\} \\ &= 2(\Delta X^2 + \Delta Y^2 + \Delta Z^2) = 2. \end{aligned}$$

From N point correspondences we can establish a matrix A in order to find the essential parameters:

$$A = \begin{bmatrix} x_1x_1' & x_1y_1' & x_1 & y_1x_1' & y_1y_1' & y_1 & x_1' & y_1' & 1 \\ x_2x_2' & x_2y_2' & x_2 & y_2x_2' & y_2y_2' & y_2 & x_2' & y_2' & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_nx_n' & x_ny_n' & x_n & y_nx_n' & y_ny_n' & y_n & x_n' & y_n' & 1 \end{bmatrix} \tag{3.34}$$

If there is no noise, from Equation 3.29 we can write $Ae=0$, where e is a nine-dimensional vector formed by stacking the elements of the matrix E into a column. With noise we will try to find a vector h such that $\|Ah\|=min$, subject to $\|h\|^2=2$.

Since E has 8 degrees of freedom, we need at least 8 point correspondences to solve for E. If we use 8 point correspondences and if the rank of A is 8, then the null space is of dimension 1 and h is the vector of norm $\sqrt{2}$ in the null space. The solution is the eigenvector of A^TA of norm $\sqrt{2}$ corresponding to the smallest eigenvalue. Then we can produce a solution; first finding the smallest eigenvalue of A^TA and then its corresponding unit eigenvector h. Then E will be $\sqrt{2}$ times this h vector. Degenerate cases occur when $Rank(A) < 8$. If 3D points are coplanar then a degenerate matrix A results.

Note that: E_t satisfies the following equations:

$$E_tE_t^T=\begin{bmatrix} \Delta Y^2+\Delta Z^2 & -\Delta Y\Delta X & -\Delta Z\Delta X \\ -\Delta Y\Delta X & \Delta Z^2+\Delta X^2 & -\Delta Y\Delta Z \\ -\Delta Z\Delta X & -\Delta Y\Delta Z & \Delta X^2+\Delta Y^2 \end{bmatrix}; \tag{3.35}$$

and also,

$$T^TE=T^TE_TR=0. \tag{3.36}$$

Since we estimated E in the previous step, now we can solve the following mean square problem in order to find T.

$$Min \ \|E^T T\|^2 \quad Subject \ \ to \quad \|T\|^2 = 1. \tag{3.37}$$

The solution is the eigenvector corresponding to the smallest eigenvalue of EE^T. We can estimate the rotation matrix R, by minimizing $\|E - E_T R\|$ with respect to R or we can find R directly from the W matrix. Let the matrix W be:

$$W = [W_1 \ \ W_2 \ \ W_3] = [E_1 x E_t + E_2 x E_3 \ \ E_2 x E_t + E_3 x E_1 \ \ E_3 x E_t + E_1 x E_2]. \tag{3.38}$$

Using the identity, $(axb)xc = (a.c)b-(b.c)a$,

$$\begin{aligned} E_1 x E_t + E_2 x E_3 &= (E_t x R_{11}) x E_t + (E_t x R_{12}) x (E_t x R_{13}) \\ &= (E_t . E_t) R_{11} - (R_{11} . E_t) E_t + (E_t . (E_t x R_{13})) R_{12} - (R_{12} . (E_t x R_{13})) E_t \\ &= R_{11} - (R_{11} . E_t) E_t + (R_{12} . (R_{13} x E_t) E_t \\ &= R_{11} - (R_{11} . E_t) E_t + ((R_{12} x R_{13}) . E_t) E_t \\ &= R_{11} - (R_{11} . E_t) E_t + (R_{11} . E_t) E_t \\ &= R_{11} \end{aligned} \tag{3.39}$$

This proves that first column of R is $E_1 x E_t + E_2 x E_3$. Similarly other columns can be found.

Using Equation 3.22 we can find the relative depths:

$$\begin{aligned} &Find \quad Z = \left(\frac{Z'}{\|T\|}, \frac{Z}{\|T\|}\right) \quad such \quad that, \\ &\left[[(x', y', 1)^T - R(x, y, 1)^T] \frac{Z'}{\|T\|} - \frac{T}{\|T\|} \right] = \min, \end{aligned} \tag{3.40}$$

using standard least square methods. The relative 3D position of any point at time t_2 and t_1 then becomes:

$$(X',Y',Z')^T=\frac{\left[R(\frac{Z}{\|T\|}(x,y,1)^T)+\frac{T}{\|T\|}+\frac{Z'}{\|T\|}(x',y',1)^T\right]}{2}; \tag{3.41}$$

$$(X,Y,Z)^T=\left[R^{-1}\left((X',Y',Z')^T-\frac{T}{\|T\|}\right)\right]. \tag{3.42}$$

C. ERROR ANALYSIS OF THE ALGORITHM

The error analysis of the algorithm will be discussed at this section using the concepts explained in Reference 2. The feature points used in the algorithm are corrupted by noise. The sources of noise are feature detector errors, matching errors, quantization errors and system calibration errors. All these errors result in errors in the solution for the motion parameters and 3D structure of the scene. The computer roundoff errors can be made small enough to be ignored for error analysis. So, the primary source of noise is assumed to be the perturbation of image coordinates.

The errors depend on the motion parameters and the scene structure. First the effects on the motion parameters and then the effects on scene structure will be discussed.

1. Motion Parameters

The error will depend the motion parameters and is discussed at this section. The magnitude and the direction of the translation vector and the rotation matrix elements will effect the errors that occur in estimating the structure and the motion of the moving object.

a. *Magnitude of Translation*

If the magnitude of translation vector is zero, the estimate of the translation direction will be arbitrary and the estimated translation vector will be an arbitrary unit vector (since $T_s \times T = 0$, where T_s is a unit vector of the same direction as the translation). From Equation 3.41 we see that the relative depths of feature points can not be determined. When $T=0$, the rank of A in Equation 3.34 can not be larger than 6 [Ref. 15]. R can still be determined in this case by picking u; any h satisfying $\|Ah\| = min$, since E is defined as $E=E_t \times R$ in Equation 3.33.

b. *Direction of Translation*

Figure 3.4 shows the relation between 3D world coordinates and the corresponding image coordinates with motion. As mentioned earlier, motion is represented by a rotation followed by a translation. From Equation 3.30 and 3.33 we can write that $x'E_tRx = 0$, where x' and x denotes image coordinates at t_2 and t_1, respectively. From this relation we can see that T is orthogonal to the cross product $x' \times Rx$. So the translation vector is determined from the relation $T.(x' \times Rx) = 0$.

Figure 3.5 shows the case where the translational vector is orthogonal to the image plane. It can be seen from the figure that the vectors $x' \times Rx$ are spread over the X-Y plane around the origin. This locus is shown by a shaded area in the figure. Figure 3.6 shows the case where the translational vector is parallel to the image plane. This time $x' \times Rx$ are confined in a small shaded area in the X-Z plane. The perturbations in the image coordinates will cause the product $x' \times Rx$ to be perturbated away from the original positions as denoted by dark disks in the figures. As explained

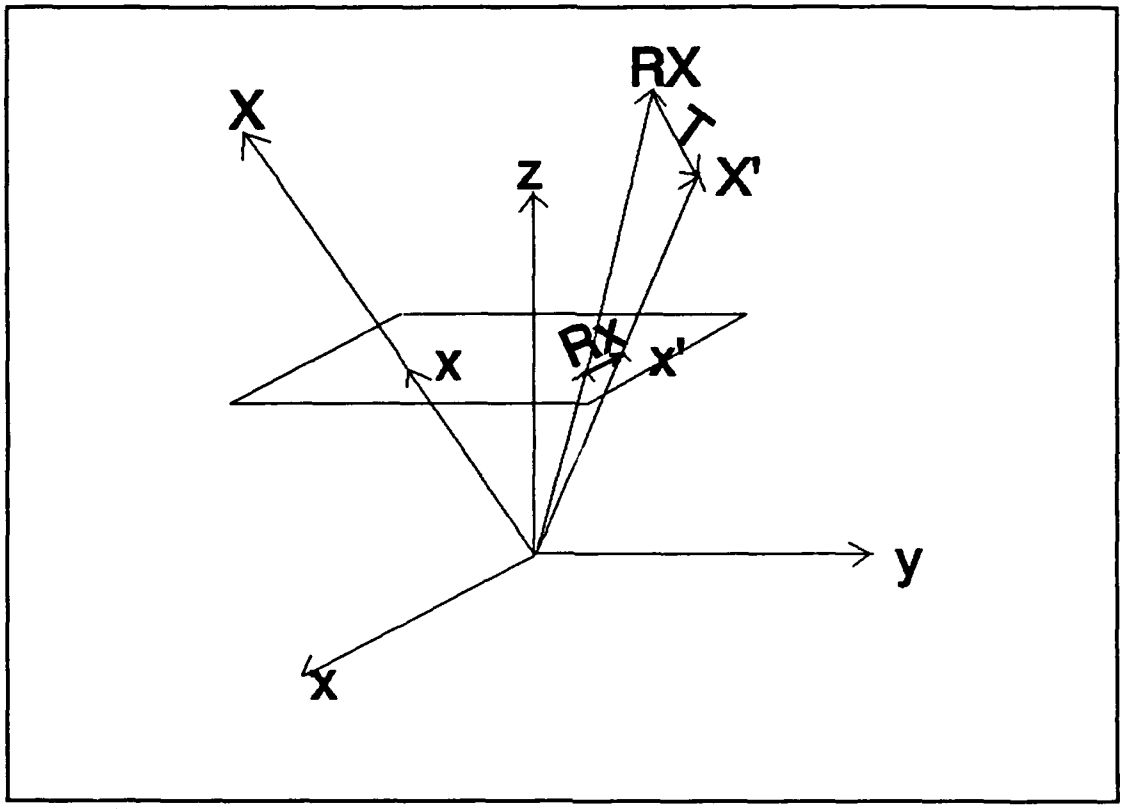

Figure 3.4 General Motion Relation

before, T is determined from $T.(x' \times Rx)=0$, so T is orthogonal to n vectors (for n correspondences) in the shaded area. Since the shaded area in Figure 3.5 spreads around the origin while the shaded area in Figure 3.6 is confined in a small area in one side of the origin, Figure 3.5 allows a more reliable estimate of T than Figure 3.6.

The perturbation of Figure 3.5 will not leave the shaded area often as of Figure 3.6. This can be seen as follows. Assume the vector x' is perturbed in the image plane. The perturbation disk is orthogonal to Rx, and almost parallel to the shaded area if Rx is near the optical axis. Similarly, the perturbation due to Rx is orthogonal to x' and so it is also nearly parallel to the shaded area if x' is near the optical axis. Hence, such individual perturbations will not cause large errors in the estimation of T.

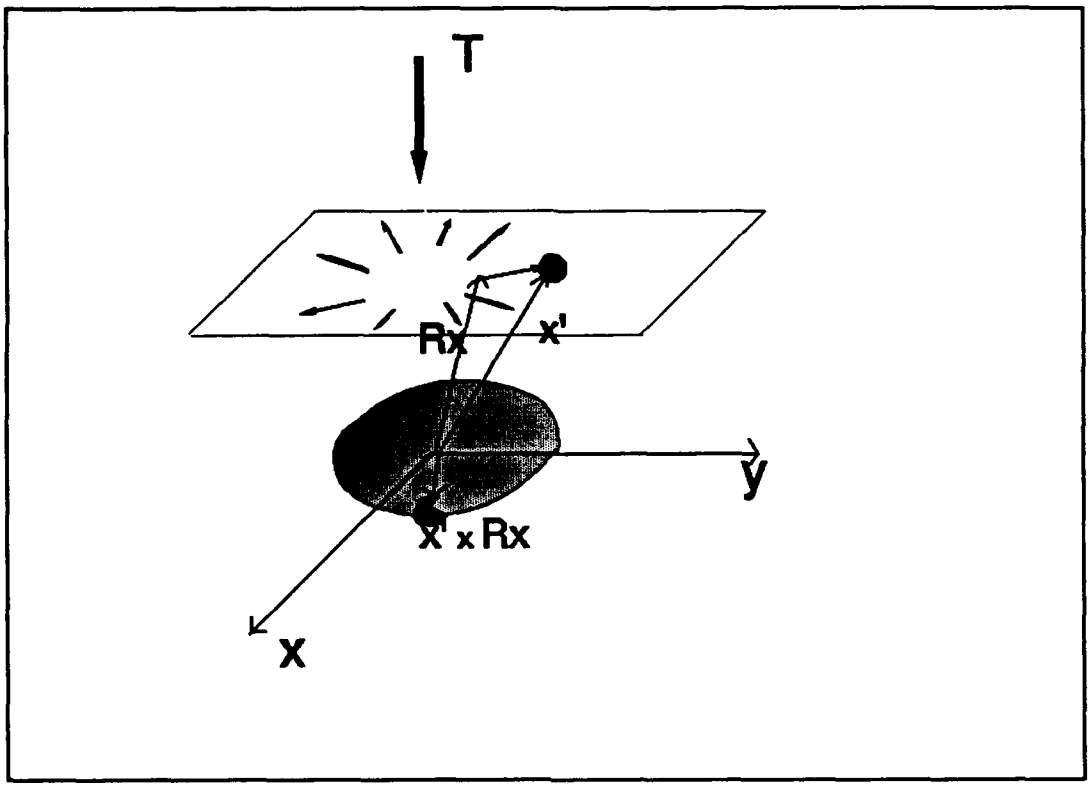

Figure 3.5 Orthogonal Translation.

In the case of Figure 3.6, the perturbation disk is nearly orthogonal to the shaded area if Rx and x' are near the optical axis as explained before. Therefore the perturbations of x' and Rx in Figure 3.6 will cause larger errors in the estimation of T than those in Figure 3.5. It is easy to see from Figure 3.6 that the largest perturbation of the estimated translation direction is in the Z component.

Both the shape of the shaded areas and the orientation of the perturbation disks imply that a translation orthogonal to the image plane allows more stable estimation of translation direction T than a translation parallel to the image plane.

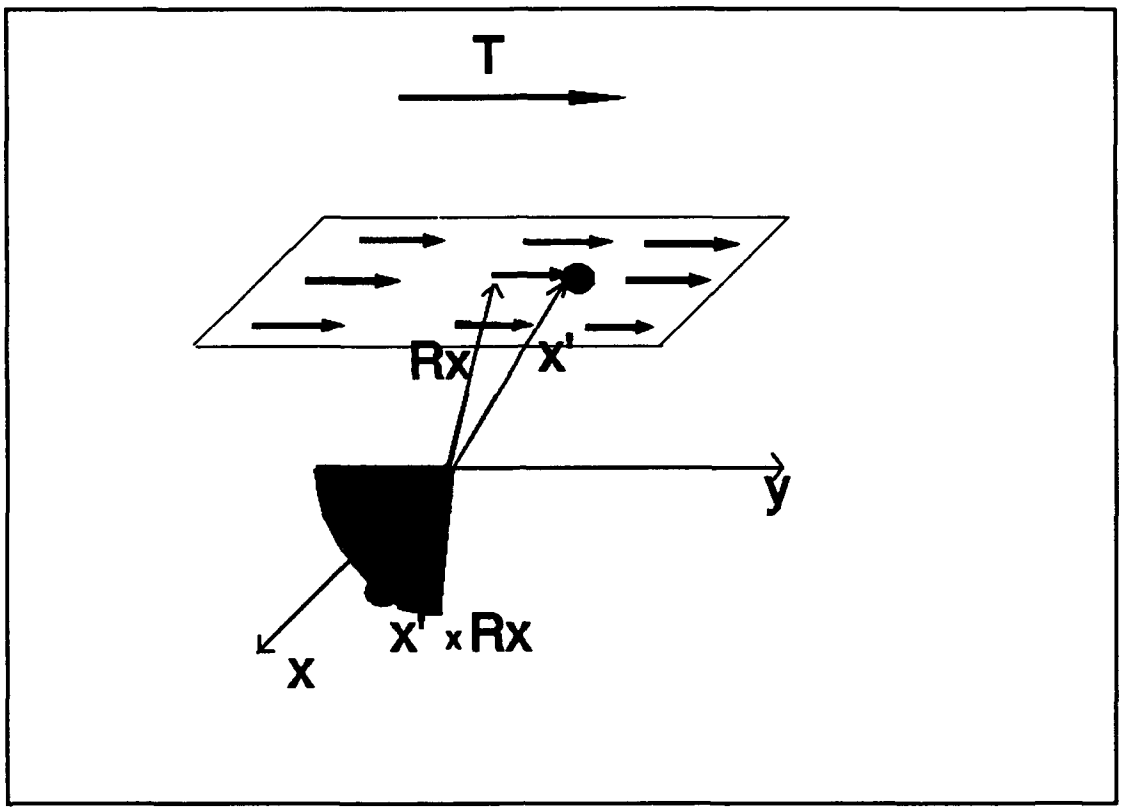

Figure 3.6 Parallel Translation.

c. *Rotation Parameters*

The correlation between the rotation and translation is very complicated. A rotation about the optical axis is easy to distinguished from translation since no translation will give the same displacement field in the image plane. But, a displacement field generated by rotation about an axis parallel to the image plane (x or y axis) may be nearly the same as that generated by a translation. The differences between these two displacement fields are not very large, especially for short displacement vectors or at the center of the images. So, the algorithm may easily confuse translation with rotation in the presence of noise. As a result, rotations with a rotation axis parallel to the image plane are more sensitive to noise than other rotations. However, since the displacement

field in the image plane is mainly caused by translation in most cases, the effects of translation are more dominant. The rotation matrix R is determined using the translation vector T (Equation 3.39). A difficult case for the estimation of translation is therefore also a difficult for the estimation of rotation.

From the above, we can conclude that long displacement vectors will generally result in more reliable solutions for rotation parameters than short ones in the presence of noise. To yield long displacement vectors, the motion should be large and/or scene should be close to image sensor.

2. Structure of the Scene

The nine-dimensional unit vector h is determined up to a sign if and only if rank of A in Equation 3.34 is equal to 8. A necessary and sufficient condition for the rank of A to equal to 8 is given by Longuet-Higgins [Ref. 15]. They define as 'degenerate' eight-point configurations for which the algorithm fails. A configuration is degenerate if as many as four of the points lie in a straight line, or if as many as seven of them lie in a plane. Degeneracy also arises if the configuration includes six points at the vertices of regular hexagon, or consists of eight points at the vertices of a cube. In the presence of noise, the rank of A is generally mathematically full even if the actual structure is degenerate. If the structure is degenerate the solution is unreliable. So, in the presence of noise, we should consider the numerical condition of the matrix A.

If the projections of feature points are confined to a small portion of the images, only a small portion of the image resolution is used. This will result in a less

reliable solution. So, the configuration of the feature points should be such that its projection covers as much of the image as possible.

In the discussion of motion parameters, we see that long displacement vectors will result in more reliable solutions. For the same amount of motion, the scene should be close to the camera so that it yields long displacement field vectors in the image plane. This condition is related to the numerical condition of matrix A.

A very effective way to reduce the error in the solutions is by using more points than the minimally required 8. Since a noise-corrupted image vector may result in a large amount of error, it is better to use only reliable feature matches for motion parameter estimation.

The resolution will affect the correctness of image coordinates. So, using high resolution will decrease the errors in the algorithm. Assuming resolution and focal length to be fixed, reducing the image size by a factor, say 2, will be equivalent to doubling the distance from the camera to the scene and reduces the variation in depth of the image. This is equivalent to reducing the resolution. So, a reduction of image size will worsen the performance of the algorithm.

The following section presents statistical data obtained through simulations that demonstrate the comments made in the discussions.

3. Experimental Results

In the simulations, the feature points are generated randomly within a cube of 10 x 10 x 10 units. The object distance is 11 units, the image size is 2 units, and the image resolution is stated for each simulation.

All the errors shown in this section are relative errors. The relative error of a matrix or a vector is defined by the Euclidean norm of the error matrix or vector divided by the Euclidean norm of the correct matrix or vector. Relative errors are often simply referred to as errors, in the following sections.

a. Simulation for Image Resolution

Figure 3.7 shows the relation between the errors in the estimates and the image resolution. It can be seen that increasing the resolution by a factor 2 roughly decreases the errors by a factor 2 which is expected from previous discussion.

b. Simulation for Number of Corresponding Points

Figure 3.8 shows the relation between the errors in the estimates and the number of corresponding points. It can been seen that an increase in the number of corresponding points will decrease the errors as expected.

c. Simulation for Image Size

In order to show the effects of decreasing image size, the image size is reduced by a factor of 2. To make sure that the same scene is visible, the object is moved away from the camera by a factor of 2. The same simulation as in the simulation for the number of corresponding points is performed again. From Figure 3.9 it can be seen that reducing the image size worsens the estimates which is consistent with earlier

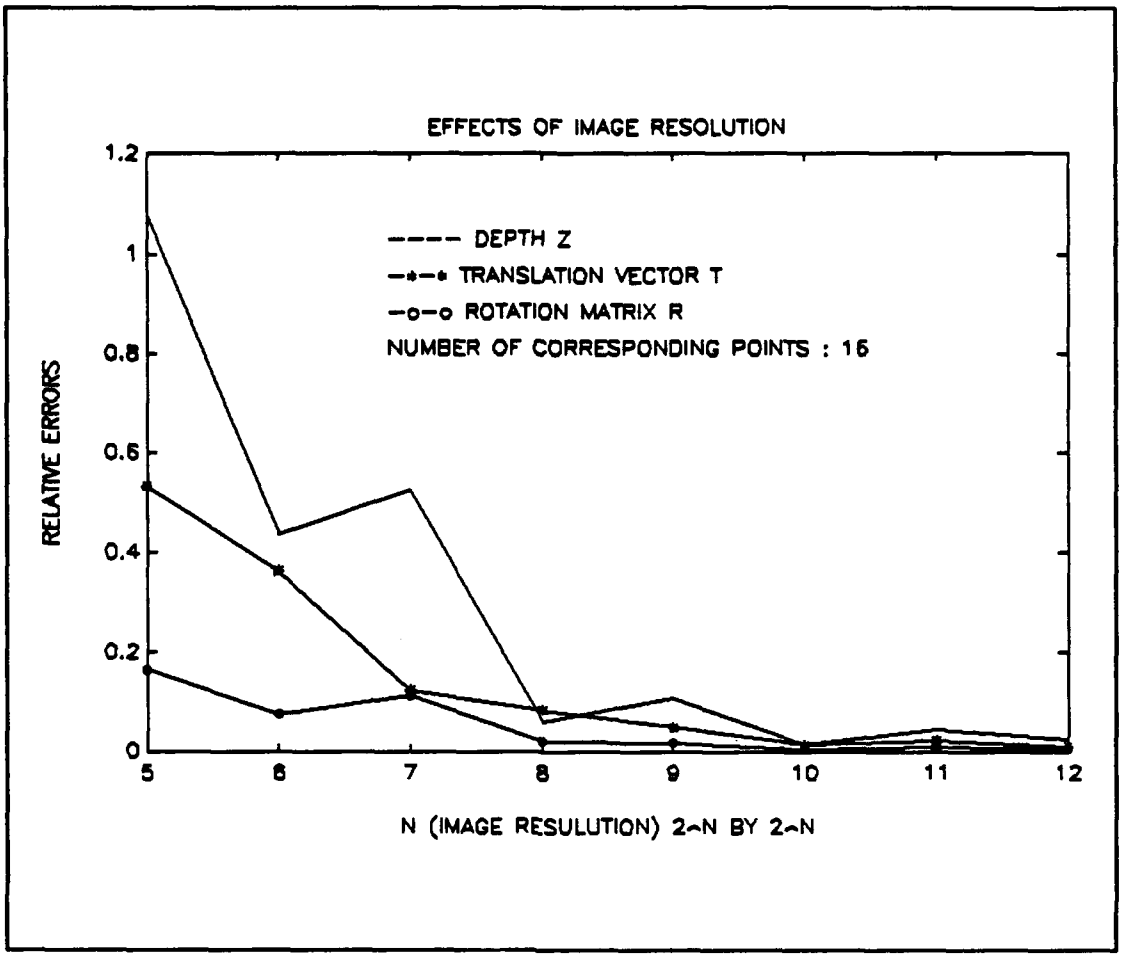

Figure 3.7 Simulation for image resolution.

discussions.

d. Simulation for Noise

In order to see the effects of perturbations in x' (second frame image coordinates) artificial image noises are added to the second frame image coordinates. The noise is added to each point in the second frame in a circle of radius r and centered at that point. This is done by perturbing the second frame image points from 0 percent to 10 percent (with 100 levels). To control the orientation, the sign of normally distributed random numbers are used. This experiment was performed several hundred

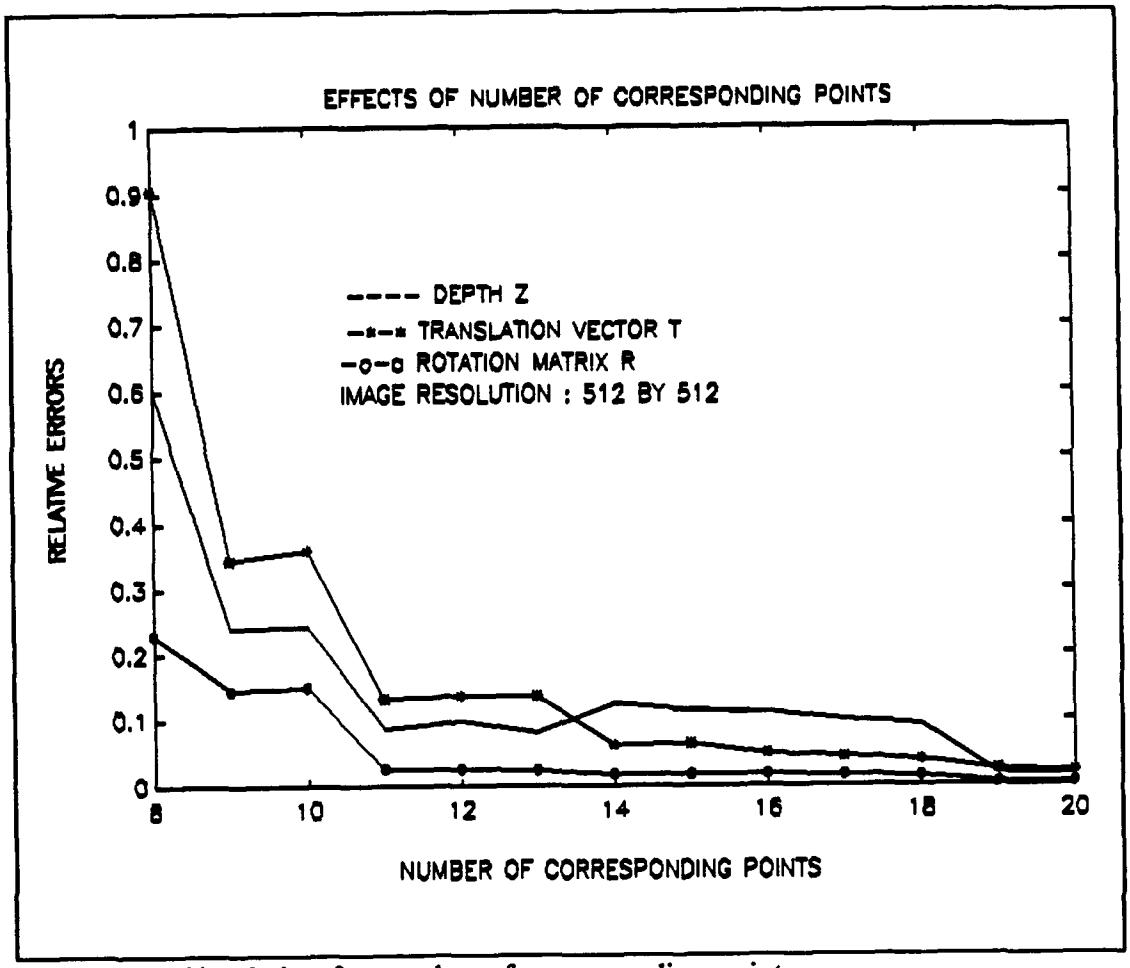

Figure 3.8 Simulation for number of corresponding points.

times. Averaging the results of these experiments Figure 3.10 is obtained.

e. Simulation for Magnitude of Translation

In order to see the effect of translation magnitude, a translation vector equal to *(t, 0, t)* is used for simulation. The value of *t* is changed from 0.5 to 4.5 with rotation axis *(1, 0, 0)* and rotation angle 8°. The results of these simulations is given in Figure 3.11. We can see that as the magnitude of translation increases, the error levels are decreased.

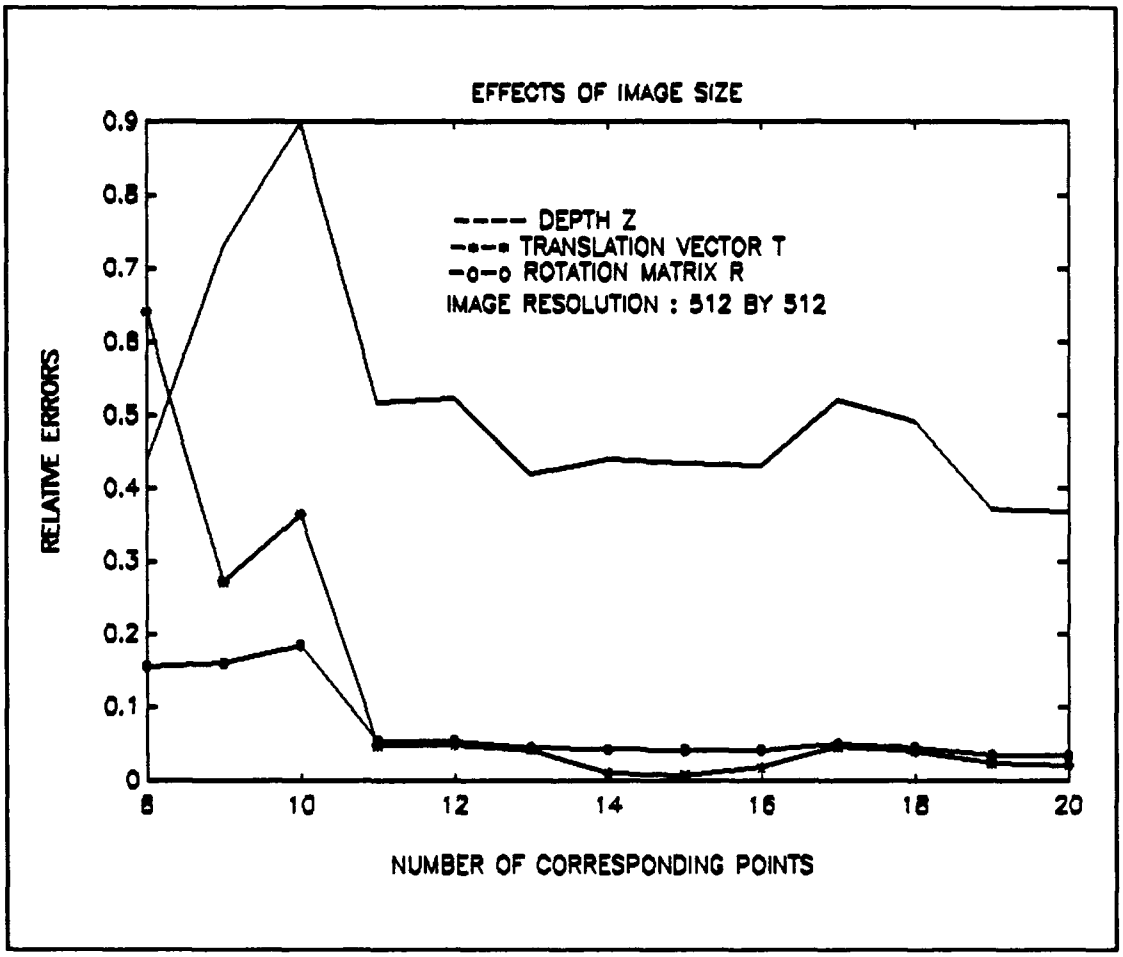

Figure 3.9 Simulation for image size.

f. Simulation for Direction of Translation

For this simulation the rotation axis is taken as *(1, 0, 0)* and the rotation angle 8°. The translation vector is taken as *(0.5, 0, k)* where *k* is changed from 0 to 2 using 20 evenly spaced values. The results of the simulations are shown in Figure 3.12. We can see that as the direction of the translation vector becomes orthogonal to the optical axis the error levels are decreased. This result is consistent with the previous discussion.

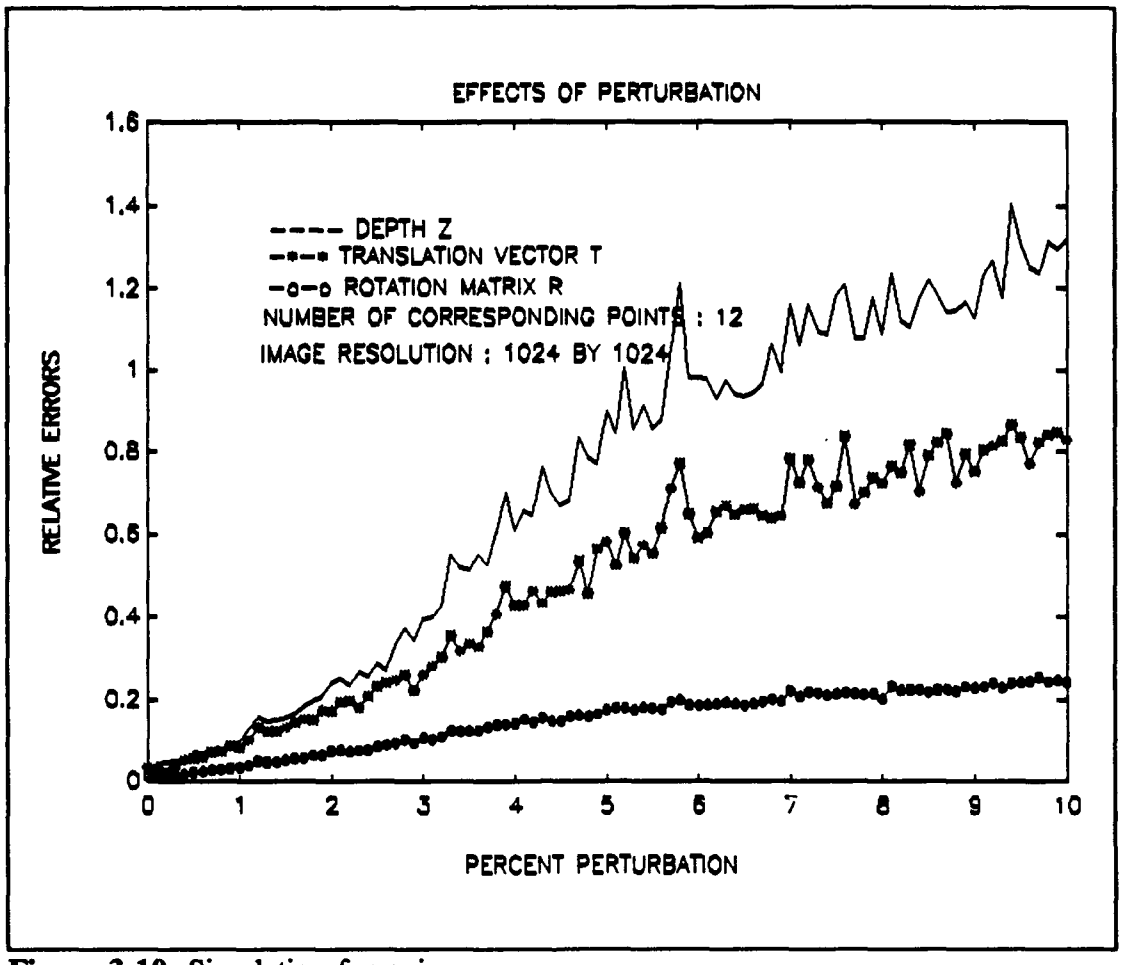

Figure 3.10 *Simulation for noise.*

D. MOTION EQUATIONS UNDER ORTHOGRAPHIC PROJECTION

We can analyze the relation between the image plane motion and the corresponding three-dimensional motion for the case of orthographic projection [Ref. 3]. The same notation and assumptions will be used as were used for the case of perspective projection.

For general rigid body motion, Equation 3.19 is still valid. From N image point correspondences:

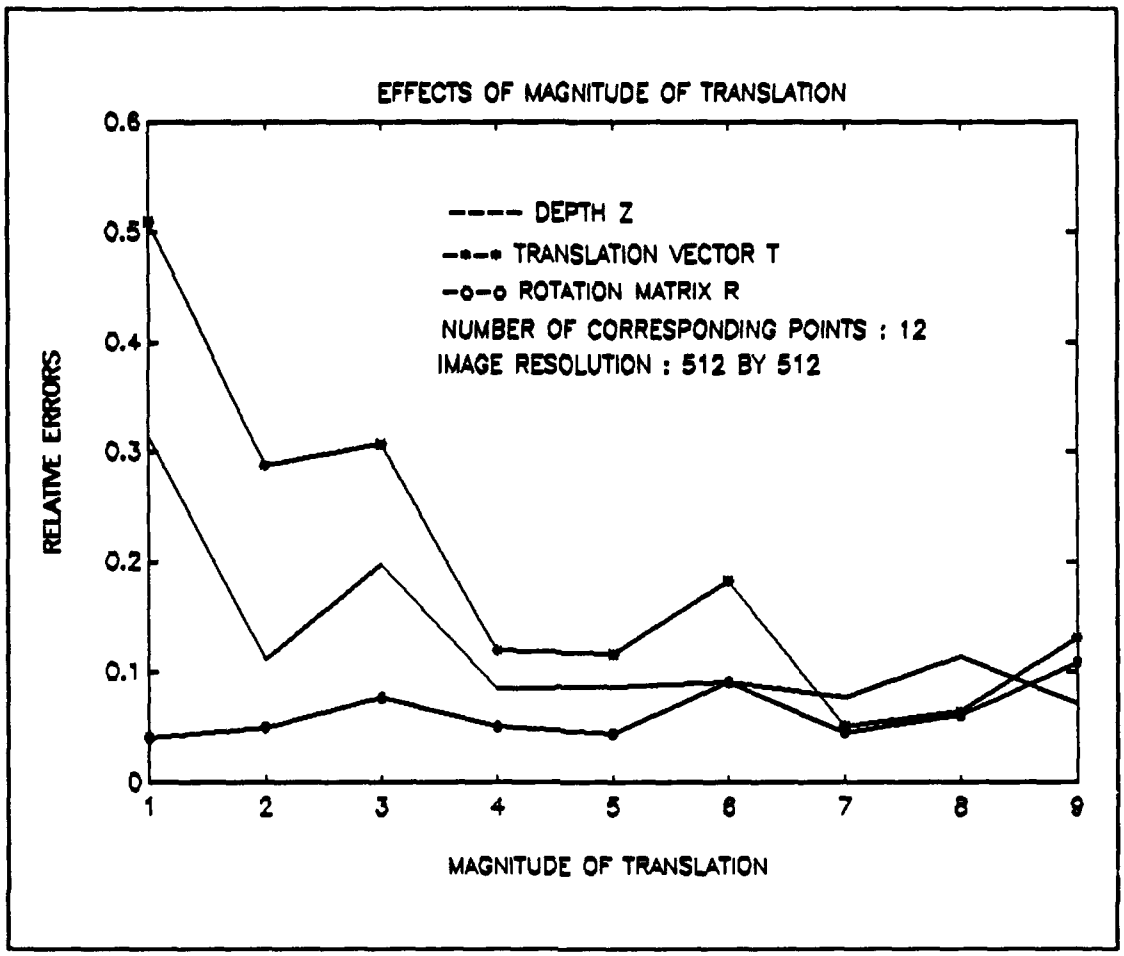

Figure 3.11 Simulation for magnitude of translation.

$$(x_i,y_i)\approx(x_i',y_i') \qquad i=1,2,\ldots,N. \tag{3.44}$$

The problem is to determine R, T, and (X_i,Y_i,Z_i) for N points. For orthographic projection from Equation 3.4:

$$\begin{aligned} x&=X, \quad y=Y; \\ x'&=X', \quad y'=Y'. \end{aligned} \tag{3.45}$$

From Equation 3.45 it is obvious that ΔZ can never be determined and the Z_i's can be determined only within an unknown additive constant. For the following discussion,

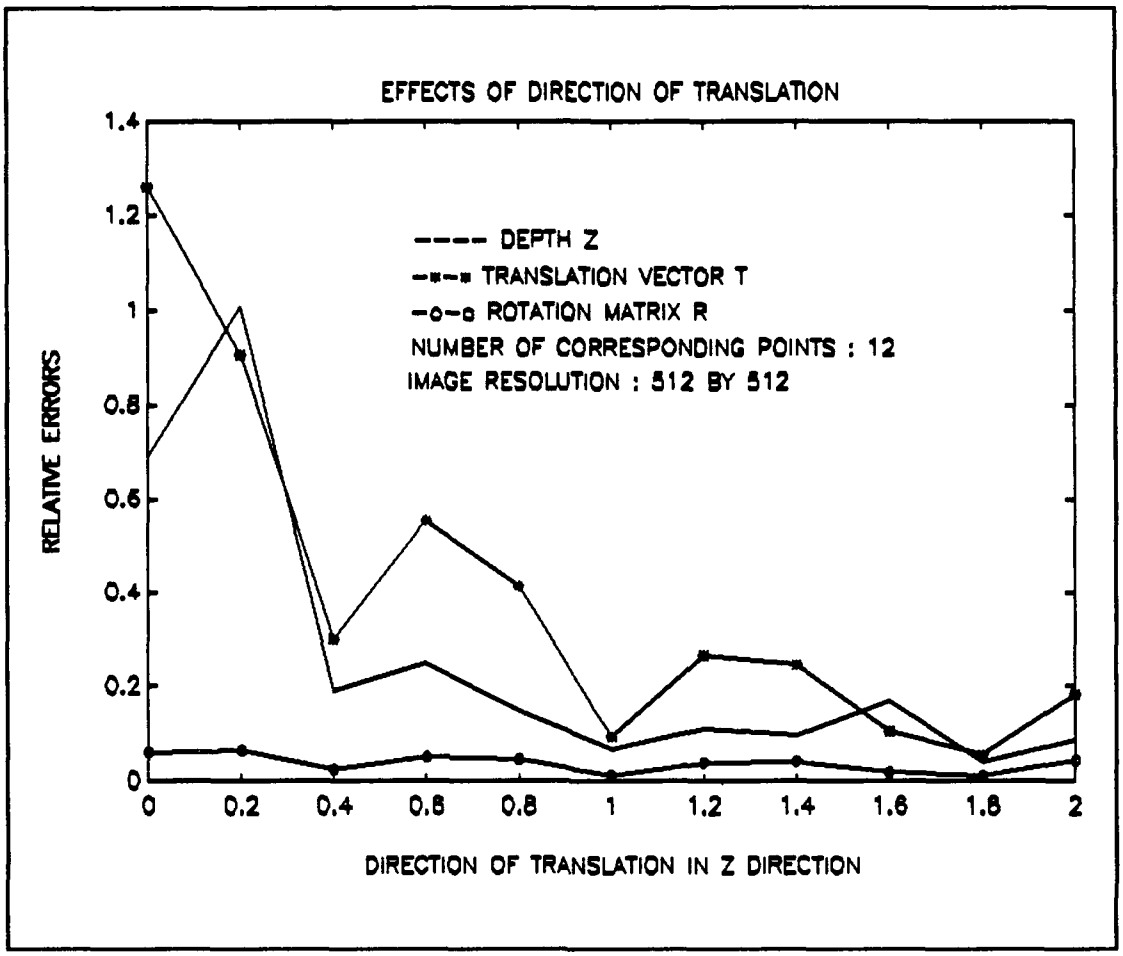

Figure 3.12 Simulation for direction of translation.

we'll try to determine: R, ΔX, ΔY, X_i, Y_i, *and* Z_i-Z_1 *for* $i=1,2,\ldots,N$.

1. Two-view Case

We assume that two orthographic views at time instants t_1 and t_2 are taken of a rigid object moving in the 3D object space with N point correspondences. We'll show that no matter how large N is, the problem has an infinite number of solutions. First, we can decompose the motion from t_1 to t_2 into a rotation R around the point (X_1,Y_1,Z_1) followed by a translation:

$$T_1 = \begin{bmatrix} X_1' - X_1 \\ Y_1' - Y_1 \\ Z_1' - Z_1 \end{bmatrix}. \tag{3.46}$$

Second, to determine R and $(X_i, Y_i, Z_i - Z_1)$, we can move (X,Y,Z) and (X',Y',Z') to the origin:

$$\begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} X_1' \\ Y_1' \\ Z_1' \end{bmatrix}. \tag{3.47}$$

Then all other points are related simply by a rotation from t_1 to t_2:

$$\begin{bmatrix} X_i' \\ Y_i' \\ Z_i' \end{bmatrix} = R \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} \qquad i = 2,3,\ldots,N. \tag{3.48}$$

From Equations 3.48 and 3.45 we can write:

$$\begin{aligned} x_i' &= r_{11}x_i + r_{12}y_i + r_{13}Z_i, \\ y_i' &= r_{21}x_i + r_{22}y_i + r_{23}Z_i. \end{aligned} \tag{3.49}$$

In matrix notation:

$$\begin{bmatrix} x_i' \\ y_i' \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} r_{13} \\ r_{23} \end{bmatrix} Z_i. \tag{3.50}$$

To eliminate Z_i, premultiply both sides of Equation 3.50 by $[r_{23}, -r_{13}]$ to get:

$$\begin{bmatrix} r_{23} & -r_{13} \end{bmatrix} \begin{bmatrix} x_i' \\ y_i' \end{bmatrix} = \begin{bmatrix} r_{23}r_{11}-r_{13}r_{21} & r_{23}r_{12}-r_{13}r_{22} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix}. \tag{3.51}$$

Using the identities:

$$\begin{aligned} r_{11}r_{23}-r_{21}r_{13} &= -r_{32}, \\ r_{12}r_{23}-r_{22}r_{13} &= r_{31}, \end{aligned} \tag{3.52}$$

which follow from the fact that each row of R is cross product of the other two, we get:

$$r_{23}x_i'-r_{13}y_i'+r_{32}x_i-r_{31}y_i = 0, \tag{3.53}$$

which is linear and homogenous in the four unknowns r_{13}, r_{23}, r_{31}, and r_{32}. From N point correspondences, we get (N-1) such equations. Therefore, if $N \geq 4$ and assuming that the points are not coplanar, we can solve the set of Equations 3.53 to obtain r_{13}, r_{23}, r_{31}, and r_{32} to within a scale factor. In the absence of noise, four point correspondences are sufficient to solve Equation 3.53, additional point correspondences are superfluous. The important point to notice here is that Equation 3.53 contains all the information we can get from the point correspondences. Therefore, no matter how many point correspondences we may have, the only thing we can determine about R is the values of r_{13}, r_{23}, r_{31}, and r_{32} to within a scale factor. Obviously, by changing the scale factor, we can get infinite number of solutions of r_{13}, r_{23}, r_{31}, r_{32} which satisfies:

$$r_{13}^2+r_{23}^2 = r_{31}^2+r_{32}^2 < 1, \tag{3.54}$$

which follows from the fact that:

$$r_{13}^2+r_{23}^2 = r_{31}^2+r_{32}^2 = 1-r_{33}^2. \tag{3.60}$$

For any of these infinitely many solutions, we can construct an R. For each solution of R, we can use Equation 3.50 to find Z_i; for $i=2,3,4$. Z_i' can be found from:

$$Z_i' = r_{31}x_i+r_{32}y_i+r_{33}Z_i. \tag{3.61}$$

We remark that as a result of the above derivation, we have a way of testing whether a set of proposed point correspondences is legitimate. Assume that we have four points over two views, then there are $4! = 24$ different mapping between the two sets of points. For each mapping, we can solve Equation 3.50 to get r_{13}, r_{23}, r_{31}, r_{32} to within a scale factor. The mapping is permissible if and only if

$$r_{13}^2+r_{23}^2 = r_{31}^2+r_{32}^2. \tag{3.62}$$

In summary, no matter how many point correspondences we have, two orthographic views result in an uncountable infinite number of solutions for motion parameters and the structure of rigid objects. We also have derived a simple test for the legitimacy of point correspondences [Ref. 3].

2. Four Points Over Three View Case

We assume that the image plane is stationary and that three orthographic views at time instants t_1, t_2, and t_3, respectively, are taken of a rigid body moving in the 3-D object space. We again use the notation explained before, except that coordinates at t_3 will be double primed, and the rotation matrix from t_2 to t_3 is denoted by:

$$S = \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{bmatrix}, \tag{3.63}$$

and the rotation from t_1 to t_3 by:

$$W = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix}. \tag{3.64}$$

Thus,

$$W = SR. \tag{3.65}$$

Assuming we are given four point correspondences over three views, again we can let:

$$\begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} = \begin{bmatrix} X_1' \\ Y_1' \\ Z_1' \end{bmatrix} = \begin{bmatrix} X_1'' \\ Y_1'' \\ Z_1'' \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}. \tag{3.66}$$

Then

$$\begin{bmatrix} X_i' \\ Y_i' \\ Z_i' \end{bmatrix} = R \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix}, \tag{3.67}$$

and

$$\begin{bmatrix} X_i'' \\ Y_i'' \\ Z_i'' \end{bmatrix} = S \begin{bmatrix} X_i' \\ Y_i' \\ Z_i' \end{bmatrix}, \tag{3.68}$$

$$\begin{bmatrix} X_i'' \\ Y_i'' \\ Z_i'' \end{bmatrix} = W \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} \quad i=2,3,4. \tag{3.69}$$

Using the method explained in the previous section, we can determine $(r_{13},r_{23},r_{31},r_{32})$, $(s_{13},s_{23},s_{31},s_{32})$ and $(w_{13},w_{23},w_{31},w_{32})$ to within scale factors:

$$\begin{aligned} (r_{13},r_{23},r_{31},r_{32}) &= \alpha(\rho_{13},\rho_{23},\rho_{31},\rho_{32}), \\ (s_{13},s_{23},s_{31},s_{32}) &= \beta(\sigma_{13},\sigma_{23},\sigma_{31},\sigma_{32}), \\ (w_{13},w_{23},w_{31},w_{32}) &= \gamma(\omega_{13},\omega_{23},\omega_{31},\omega_{32}). \end{aligned} \tag{3.70}$$

where ρ_{ij}, σ_{ij}, and ω_{ij} are known α, β, and γ are unknown constants which are assumed to be nonzero.

Equation 3.65 is the only constraint we have on r_{ij}, s_{ij}, and w_{ij}. We can rewrite it:

$$\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} = \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{32} \\ s_{31} & s_{32} & s_{33} \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}. \tag{3.71}$$

Multiplying out, we get:

$$\begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} = \begin{bmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} + \begin{bmatrix} s_{13} \\ s_{23} \end{bmatrix} \begin{bmatrix} r_{31} & r_{32} \end{bmatrix}, \tag{3.72}$$

$$\begin{bmatrix} w_{13} \\ w_{23} \end{bmatrix} = \begin{bmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{bmatrix} \begin{bmatrix} r_{13} \\ r_{23} \end{bmatrix} + \begin{bmatrix} s_{13} \\ s_{23} \end{bmatrix} r_{33}, \tag{3.73}$$

and

$$\begin{bmatrix} w_{31} & w_{32} \end{bmatrix} = \begin{bmatrix} s_{31} & s_{32} \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} + s_{33} \begin{bmatrix} r_{31} & r_{32} \end{bmatrix}, \tag{3.74}$$

$$w_{33} = \begin{bmatrix} s_{31} & s_{32} \end{bmatrix} \begin{bmatrix} r_{13} \\ r_{23} \end{bmatrix} + s_{33} r_{33}. \tag{3.75}$$

From Equations 3.73 and 3.74, we can determine α, β, and γ. Then, we can find R and S. Premultiplying both sides of Equation 3.73 by $[s_{23}, -s_{13}]$, we get:

$$\begin{bmatrix} s_{23} & -s_{13} \end{bmatrix} \begin{bmatrix} w_{13} \\ w_{23} \end{bmatrix} = \begin{bmatrix} s_{23}s_{11} - s_{13}s_{21} & s_{23}s_{12} - s_{13}s_{22} \end{bmatrix} \begin{bmatrix} r_{13} \\ r_{23} \end{bmatrix}, \tag{3.76}$$

or,

$$\begin{bmatrix} s_{23} & -s_{13} \end{bmatrix} \begin{bmatrix} w_{13} \\ w_{23} \end{bmatrix} = \begin{bmatrix} -s_{32} & s_{31} \end{bmatrix} \begin{bmatrix} r_{13} \\ r_{23} \end{bmatrix}. \tag{3.77}$$

From Equation 3.70:

$$\beta\gamma(\sigma_{23}\omega_{13} - \sigma_{13}\omega_{23}) = \beta\alpha(-\sigma_{32}\rho_{13} + \sigma_{31}\rho_{23}), \tag{3.78}$$

and

$$\frac{\alpha}{\gamma}=\frac{\sigma_{23}\omega_{13}-\sigma_{13}\omega_{23}}{-\sigma_{32}\rho_{13}+\sigma_{31}\rho_{23}}. \tag{3.79}$$

Similarly, postmultiplying both sides of Equation 3.74 by $[r_{32},-r_{31}]^T$ yields:

$$\frac{\beta}{\gamma}=\frac{\omega_{31}\rho_{32}-\omega_{32}\rho_{31}}{-\sigma_{31}\rho_{23}+\sigma_{23}\rho_{13}}. \tag{3.80}$$

Thus, α/γ and β/γ are determined if:

$$r_{13}s_{32}-r_{23}s_{31}\neq 0. \tag{3.81}$$

Next, premultiplying both sides of Equation 3.73 by $[s_{13},s_{23}]$, we get:

$$\begin{bmatrix} s_{13} & s_{23} \end{bmatrix}\begin{bmatrix} w_{13} \\ w_{23} \end{bmatrix} = \begin{bmatrix} s_{13}s_{11}+s_{23}s_{21} & s_{13}s_{12}+s_{23}s_{22} \end{bmatrix}\begin{bmatrix} r_{13} \\ r_{23} \end{bmatrix}+(s_{13}^2+s_{23}^2)r_{33}, \tag{3.82}$$

or,

$$\begin{bmatrix} s_{13} & s_{23} \end{bmatrix}\begin{bmatrix} w_{13} \\ w_{23} \end{bmatrix} = \begin{bmatrix} -s_{33}s_{31} & -s_{33}s_{32} \end{bmatrix}\begin{bmatrix} r_{13} \\ r_{23} \end{bmatrix}+(s_{13}^2+s_{23}^2)r_{33}. \tag{3.83}$$

From Equation 3.70 we get:

$$\beta\alpha(\sigma_{13}\omega_{13}+\sigma_{23}\omega_{23}) = -s_{33}\beta\alpha(\sigma_{31}\rho_{13}+\sigma_{32}\rho_{23})+\beta^2(\sigma_{13}^2+\sigma_{23}^2)r_{33}, \tag{3.84}$$

or,

$$(\sigma_{13}\omega_{13}+\sigma_{23}\omega_{23}) = -\frac{\alpha}{\gamma}(\sigma_{31}\rho_{13}+\sigma_{32}\rho_{23})s_{33}+\frac{\beta}{\gamma}(\sigma_{13}^2+\sigma_{23}^2)r_{33}. \tag{3.85}$$

Similarly, postmultiplying both sides of Equation 3.74 by $[r_{31},r_{32}]^T$ yields:

$$(\omega_{31}\rho_{31}+\omega_{32}\rho_{32}) = -\frac{\beta}{\gamma}(\sigma_{31}\rho_{13}+\sigma_{32}\rho_{23})r_{33}+\frac{\alpha}{\gamma}(\rho_{31}^2+\rho_{32}^2)s_{33}. \tag{3.86}$$

A unique solution for $(\beta/\gamma)r_{33}$ and $(\alpha/\gamma)s_{33}$ can be determined from Equations 3.85 and 3.86 if:

$$\begin{bmatrix} (s_{13}^2+s_{23}^2) & -(s_{31}r_{13}+s_{32}r_{23}) \\ -(s_{31}r_{13}+s_{32}r_{23}) & (r_{31}^2+r_{23}^2) \end{bmatrix} \neq 0. \tag{3.87}$$

Then, since α/γ and β/γ have already been obtained, we can determine r_{33} and s_{33} uniquely.

From:

$$r_{13}^2+r_{23}^2+r_{33}^2 = 1, \tag{3.88}$$

and from Equation 3.70, we have:

$$\alpha^2(\rho_{13}^2+\rho_{23}^2) = 1-r_{33}^2, \tag{3.89}$$

or

$$\alpha^2 = \frac{1-r_{33}^2}{\rho_{13}^2+\rho_{23}^2}. \tag{3.90}$$

Thus, we have two solutions for α:

$$\alpha = \pm\sqrt{\frac{1-r_{33}^2}{\rho_{13}^2+\rho_{23}^2}}. \tag{3.91}$$

From Equations 3.79 and 3.80, we have two solutions to α, β, and γ. Then, we have two solutions for r_{13}, r_{23}, r_{31}, r_{32}, r_{33}; s_{13}, s_{23}, s_{31}, s_{32}, s_{33}, at this moment. For each solution, we can determine the remaining elements of R and S by the method described below. To find r_{11} and r_{12}, we may use two properties of the rotation matrix:

$$r_{31}r_{11}+r_{32}r_{12} = -r_{33}r_{13}, \tag{3.92}$$

$$r_{32}r_{11}-r_{31}r_{12} = -r_{23}. \tag{3.93}$$

Equations 3.92 and 3.93 denotes two linear equations with two unknowns (r_{11} and r_{12}), and the coefficient matrix has the determinant:

$$\begin{vmatrix} r_{31} & r_{32} \\ r_{32} & -r_{31} \end{vmatrix} = -r_{31}^2-r_{32}^2, \tag{3.94}$$

which is assumed to be nonzero. Therefore we obtain a unique solution for r_{11}, r_{12}. Similarly, a unique solution for r_{21} and r_{22} can be obtained. Using the same method, s_{11}, s_{12}, s_{21}, s_{22} can be found.

Using Equation 3.50, the Z_i's can be determined for $i=2,3,4$. As pointed out in [Ref. 10], four point correspondences over three frames yield two solutions for motion and structure. The point configurations of these two solutions are reflections of each other with respect to the image plane.

E. SIMULATION

Computer generated random numbers are used as feature points when simulating the algorithm. The second and the third frame image points are perturbed within a circle with radius equal to a varying percent of the original values. This experiment is performed several hundred times and the results are plotted on Figure 3.13.

From Figure 3.13 we can see that the algorithm is very sensitive to the perturbation of the feature points. Even 0.1 percent perturbation is enough to obtain very large errors. So the algorithm is valid only for ideal cases and can not be used with low SNR images.

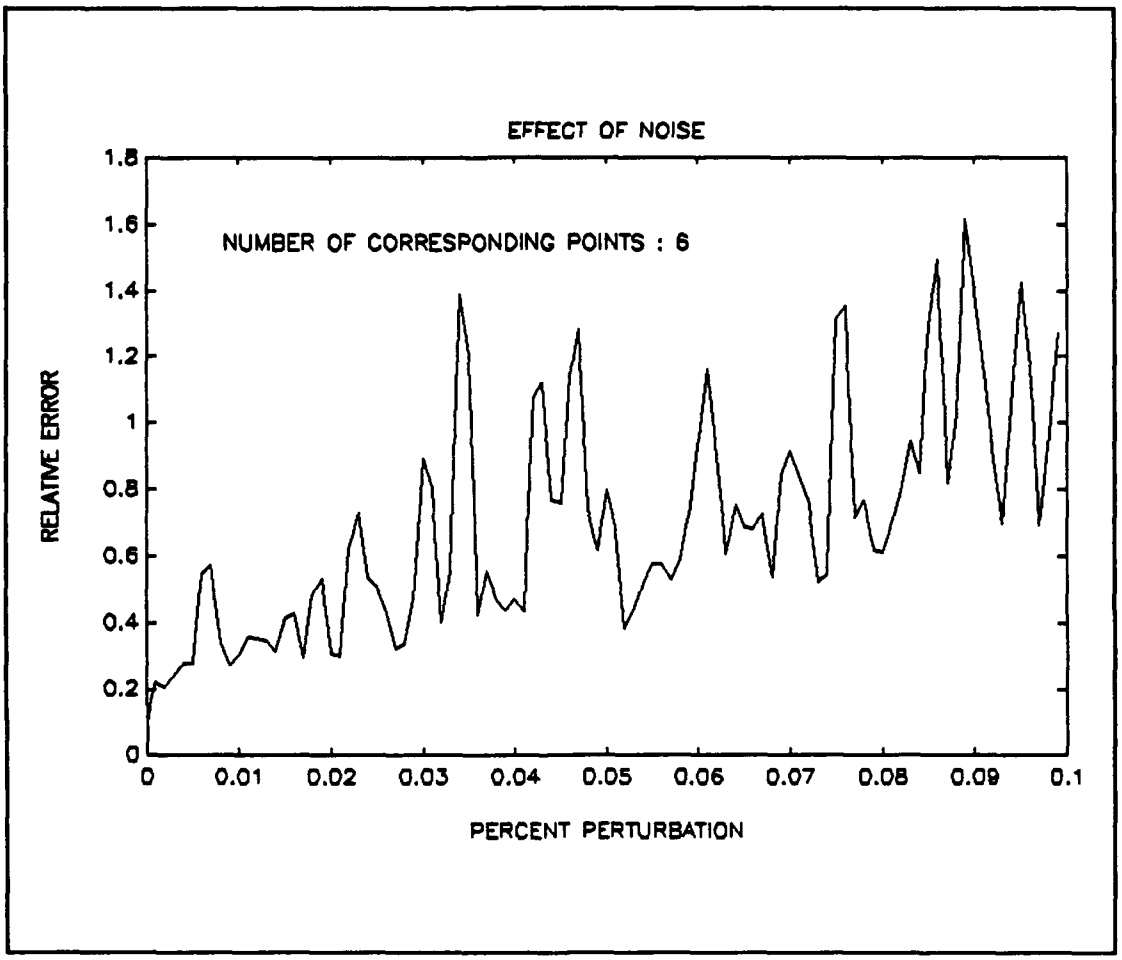

Figure 3.13 Simulation for perturbation.

IV. DIFFERENCE IMAGES

A. GENERAL

One of the simplest approaches for detecting the changes between two image frames taken at different times, is to compare these two images on a pixel by pixel basis. One way of doing this is to form a difference image (DI). A difference image is a binary image generated by comparing the two frames. The DI is generated by placing "1" in those pixel positions for which the corresponding pixels in the two frames being compared have an appreciable difference in their gray level characteristics. This operation is stated as:

$$f_d(x,y,t_i,t_j) = \begin{cases} 1 & \text{if } |f(x,y,t_i) - f(x,y,t_j)| > T, \\ 0 & \text{Otherwise.} \end{cases} \tag{4.1}$$

where $f(x,y,t_i)$, $f(x,y,t_j)$ are image frames at times t_i and t_j respectively, $f_d(x,y,t_i,t_j)$ is the difference image and T is a threshold. The operation is computationally straightforward because it involves only the subtraction of corresponding pixels. This approach is applicable only if the illumination is relatively constant within the bounds established by threshold.

Difference images alone reveal little information as to the higher level nature of scene and sensor change as reflected in the image plane. For example, the difference images will reflect the combination of motion effects in the case of several moving objects [Ref. 16].

Difference images are very vulnerable to noise. 1-valued entries in f_d in Equation 4.1 often arise as a result of noise. The removal of noise may be achieved by forming 4 or 8 connected regions of 1s in f_d and then ignoring any region that has less than a predetermined number of entries. Unfortunately, this results in ignoring small and/or slow-moving objects [Ref. 17].

The concepts explained above are illustrated in Figures 4.1 and 4.2. Figure 4.1 shows an image frames taken at times t_i and t_j containing a single object of constant

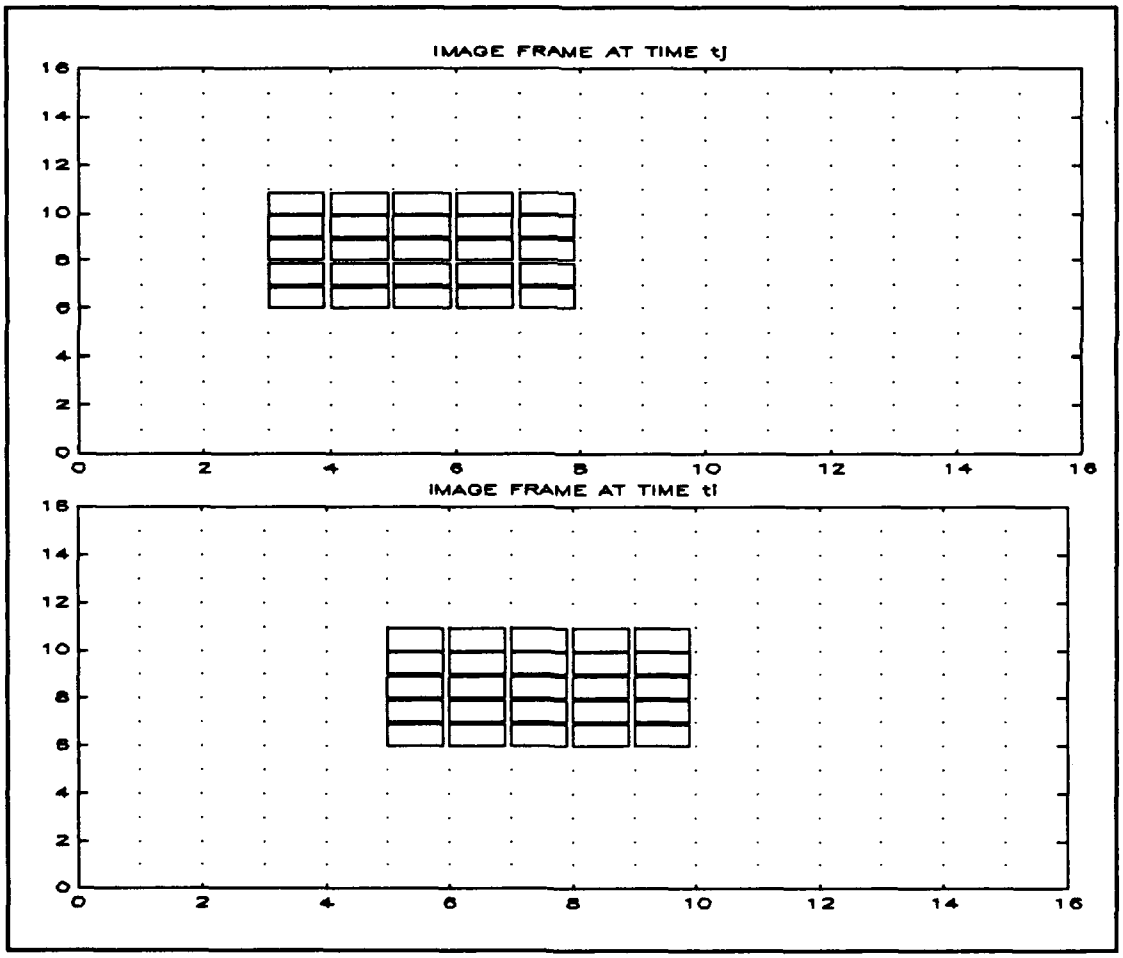

Figure 4.1 Image frames at times t_j and t_i.

intensity denoted by □s. Figure 4.2 shows the difference image computed using Equation 4.1 with a threshold larger than the constant background intensity. Note that two disjoint regions R_1 and R_2 in the difference image are generated. The region of intensity R_1 which is called the *leading edge* is generated because of "covering" the background and R_2 which is called the *trail edge* is generated because of "uncovering" the background over time interval t_j-t_i. The region between R_1 and R_2 is zero because of

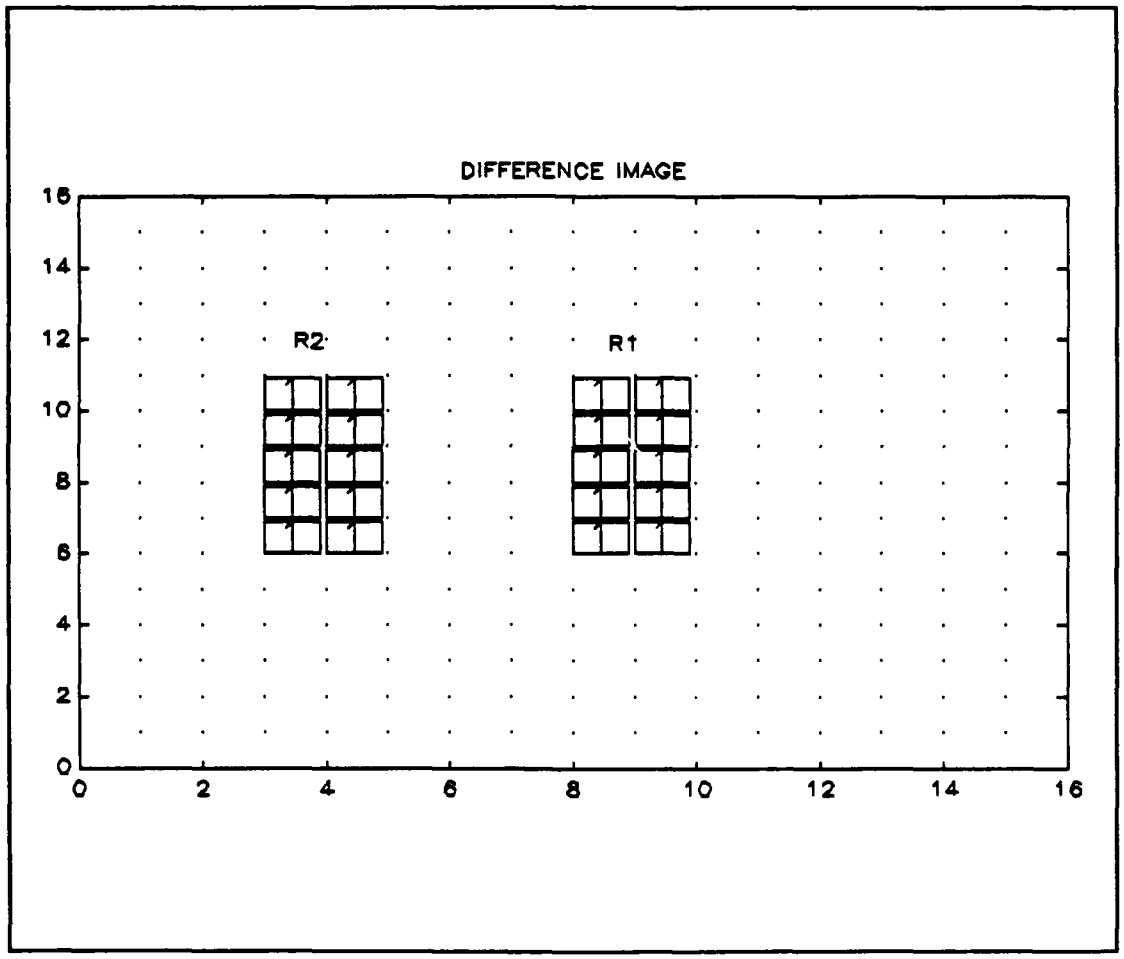

Figure 4.2 Difference Image.

the presence of the object in this region in both images and the object has constant

intensity. Since the background is assumed to be unchanged in both images the background region will also be zero in the difference image.

B. ACCUMULATIVE DIFFERENCES

A sequence of images $f(x,y,t_1), f(x,y,t_2), \ldots, f(x,y,t_n)$ are taken at times $t_1, t_2, \ldots, t_n$ respectively and let $f(x,y,t_1)$ be the reference image. An accumulative difference image is formed by comparing the reference with every subsequent image in the sequence. A counter for each pixel location is incremented every time there is a difference at that pixel location between the reference and an image in the sequence. Thus, when the kth frame is being compared with the reference, the entry in a given pixel of the accumulative image gives the number of times the gray level at that position has been different from the corresponding pixel value in the reference image. At each time, differences are established by using Equation 4.1 then they are summed up to generate the accumulative image.

These concepts are illustrated in Figure 4.3. A rectangular object has been placed at the upper left corner of the picture then moved to the right and down at a constant velocity of 1 pixel/frame (column velocity) and 2 pixel/frame (row velocity). Figure 4.3 shows the corresponding accumulative difference images for the 2nd and 4th frames respectively. The velocity of the object can be estimated from the repetition times of the homogeneous rows (every element in the row is equal to each other) and homogeneous columns (every element in the column is equal to each other) in the accumulative image. From Figure 4.3 we see that there are one homogeneous column and two homogeneous rows at each time. This shows that the column velocity is 1 and row velocity is 2.

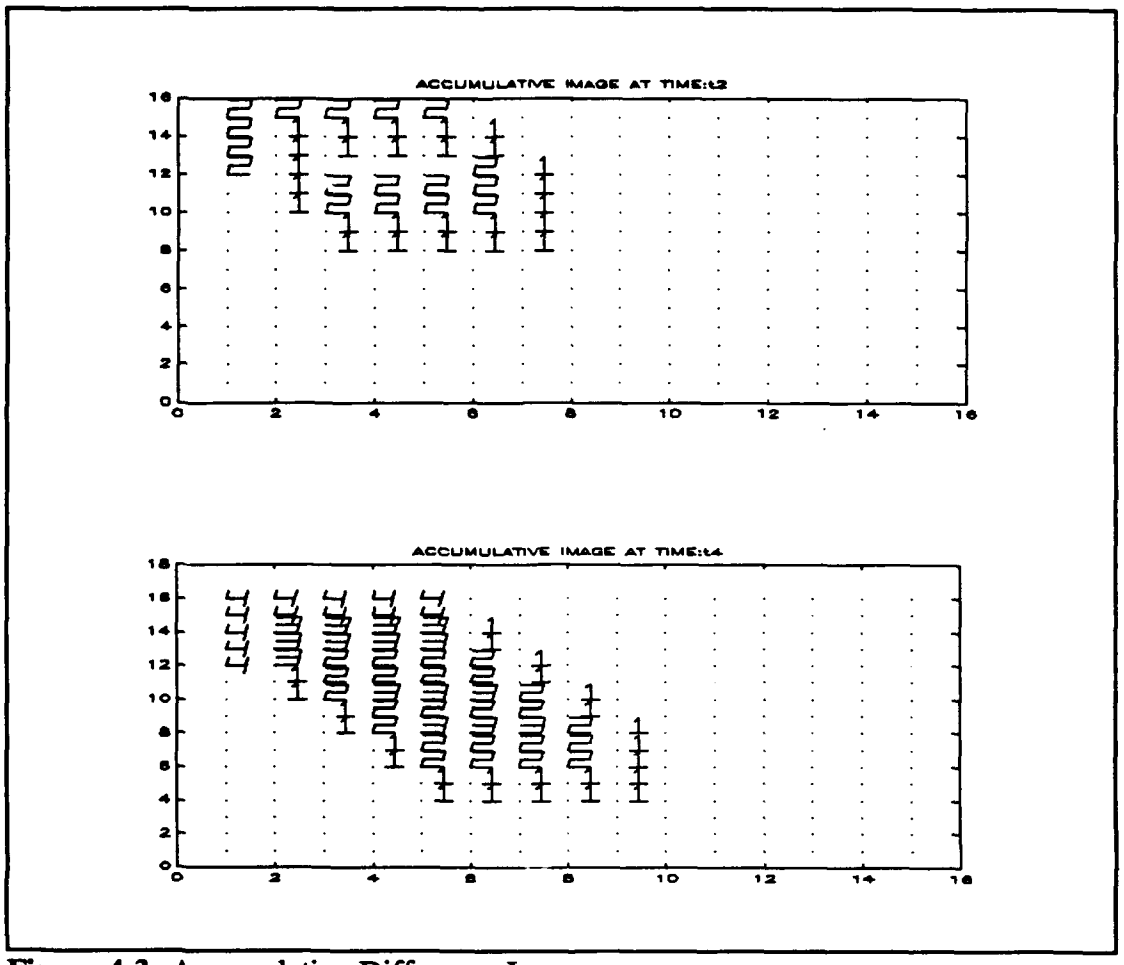

Figure 4.3 Accumulative Difference Images.

Finding velocity from the accumulative image becomes very hard if noise is present in the image since some pixel counters may be incremented because of the noise. This effect can be reduced using 4 or 8 connected regions as explained previously. Also, it can be seen from Equation 4.1 that selecting the proper threshold value is very important for estimating the correct object velocities. This is illustrated in simulation section.

The object velocity can also be estimated from the changes in the center of the area that the object covers at each picture. This can be accomplished by obtaining binary images for each frame. This binary image algorithm is explained in the next section.

C. BINARY IMAGES AND VELOCITY ESTIMATION

Binary (two-valued or black-and-white) images are obtained by setting pixel values to "0" for all image points corresponding to the background and setting pixel values to "1" for all the image points on the object. If the object appears consistently darker (or brighter) than the background, it is easy to generate the binary images, since generating binary images simply requires thresholding the pixel values. If the object has similar gray-level values as the background and/or noise is corrupting the image, the thresholding procedure becomes harder. As explained in Reference 5, Histogramming or Segmentation methods should be applied in this case to obtain the binary image.

Assuming the image has been digitized with n rows and m columns, the area of the object can be computed in units of the area of a picture cell:

$$A=\sum_{i=1}^{n}\sum_{j=1}^{m}p_{ij}, \tag{4.2}$$

where p_{ij} is the value of binary image at the ith row and j^{th} column. The x-position of center of the area (COA) can be found by using:

$$A_x = \frac{1}{A}\sum_{i=1}^{m} ik_i, \tag{4.3}$$

where k_i is the x-projection of the image, which gives for each column the number of picture cells in the column that have the value one. Similarly, the y-position of the COA can be found using:

$$A_y = \frac{1}{A}\sum_{j=1}^{n} jh_j, \tag{4.4}$$

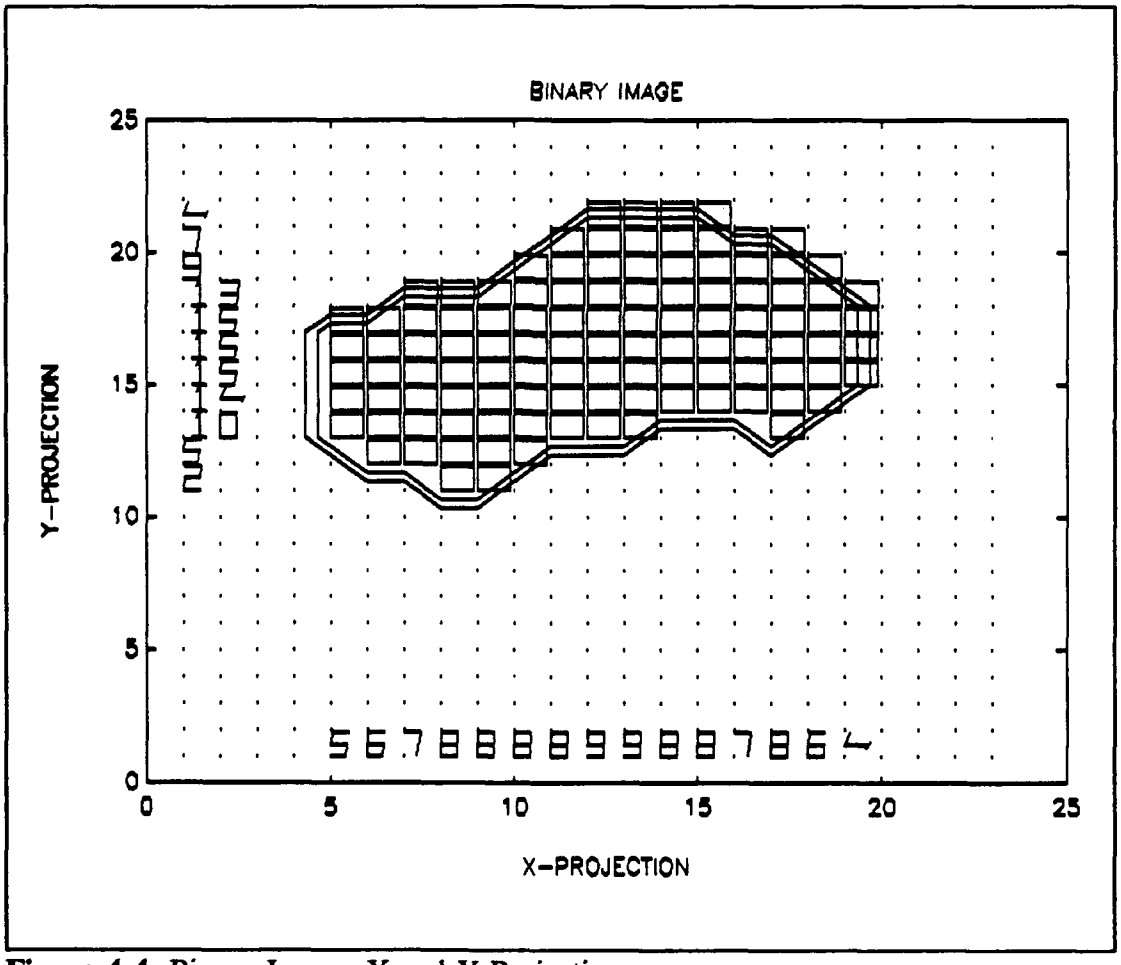

Figure 4.4 Binary Image, X and Y Projections.

where h_j is the y-projection of the image, which gives for each row the number of picture cells in the row that have the value one. These concepts are illustrated in Figure 4.4.

Assuming the object is rigid and each image is generated by using orthographic projection, the x-component of the object velocity can be found simply from:

$$v_x = \frac{A_{x2} - A_{x1}}{t_2 - t_1}, \tag{4.5}$$

where A_{x2}, A_{x1} are the x-components of COAs at time t_2 and t_1, respectively. Similarly, y-component of the velocity can be found from:

$$v_y = \frac{A_{y2} - A_{y1}}{t_2 - t_1} \tag{4.6}$$

where A_{y2} and A_{y1} are the y-components of COAs at time t_2 and t_1, respectively.

Using the algorithms explained in this chapter, some experiments has been performed with real images and computer generated images. The simulations and the results are presented in the next section.

D. SIMULATION

The concepts explained for accumulative differences are applied to a computer generated square object. The object has been moved at a rate of 5 pixels/frame right and 5 pixels/frame down. The accumulative difference for 8 frames is obtained as explained above. The resultant image is shown on Figure 4.5 from which it is possible to estimate the object velocity since no noise is added and the object has a regular shape. If the image has noise, and/or has irregular shape, it is very hard to estimate the velocity of the object. To demonstrate this the car image is used which is shown in Figure 4.6.

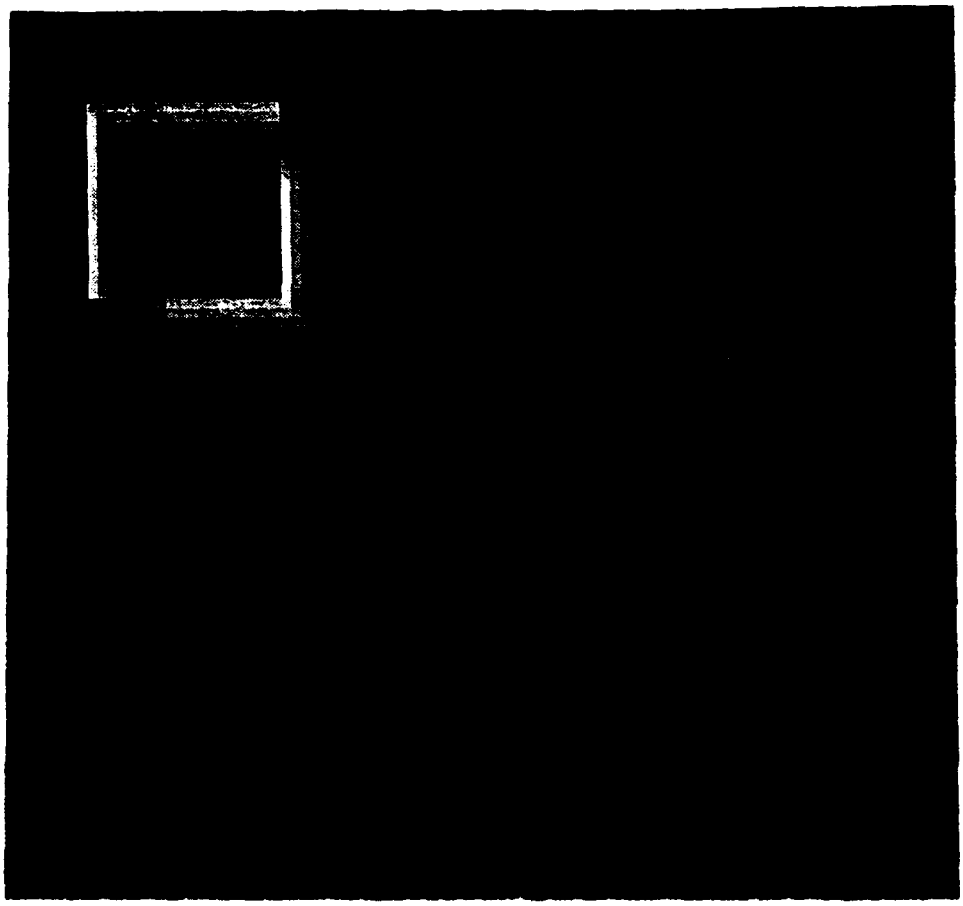

Figure 4.5 Computer generated accumulative difference image.

The accumulative difference for this image for 3 frames is shown on Figure 4.7. Since the background is constant for three frames, it is eliminated except for the regions the car covers or uncovers. Velocity estimation is very hard in that case. The only thing we can obtain is the type of motion, [Ref. 17]. So, we can say that this algorithm is adequate only for an "early warning system" which detects only changes and the direction of the motion.

The concepts explained for binary images are simulated using computer generated binary images. A square object has been moved through the image plane and artificial

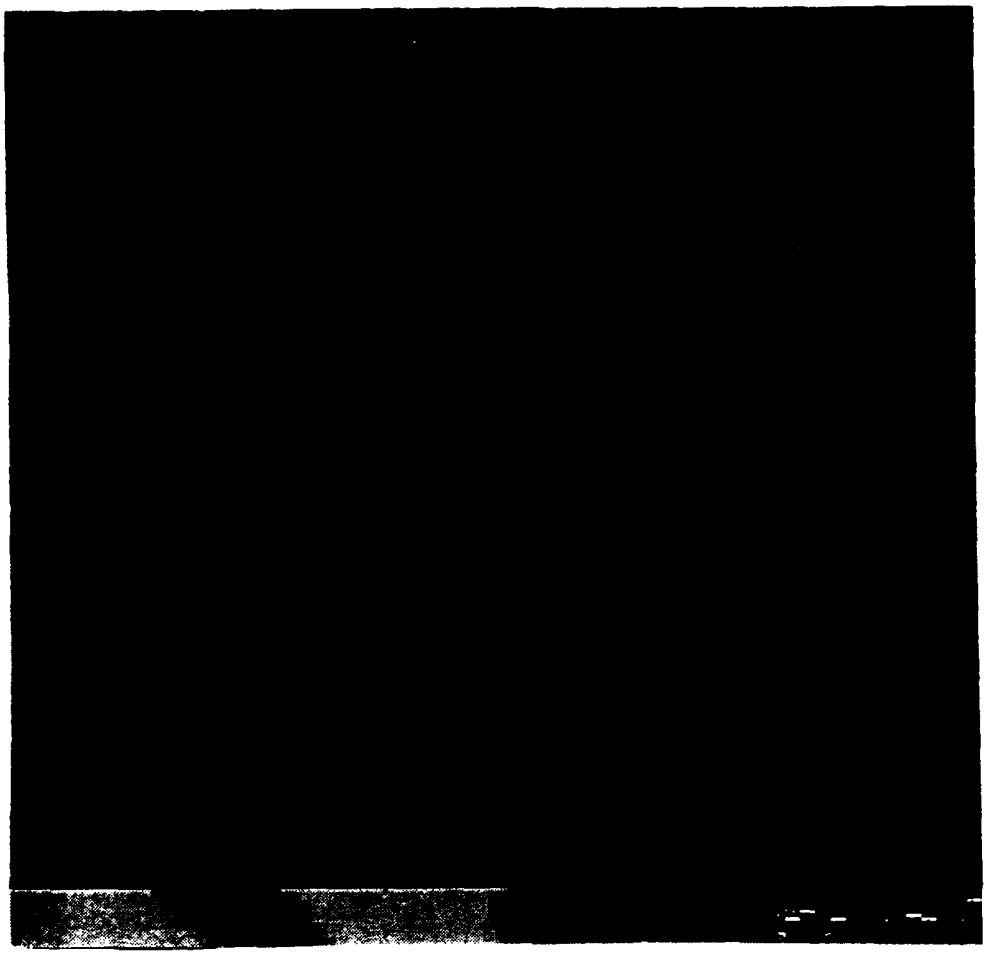

Figure 4.6 Car image.

image noise is added at each frame. To reduce the effect of noise, a thresholding process is applied to the frames, then only nonzero four-connected pixel regions are used to estimate the motion using the changes in the COA. The relative errors for different SNR are obtained. Figure 4.8 shows the result of the first experiment with a threshold of 0.5. Figure 4.9 shows the result of another experiment with a threshold 0.25.

From Figures 4.8 and 4.9, we can see that as we increase the threshold the relative error becomes smaller, but we may ignore the small changes and slow moving objects which is consistent with the previous discussion. The error levels for high SNR may be

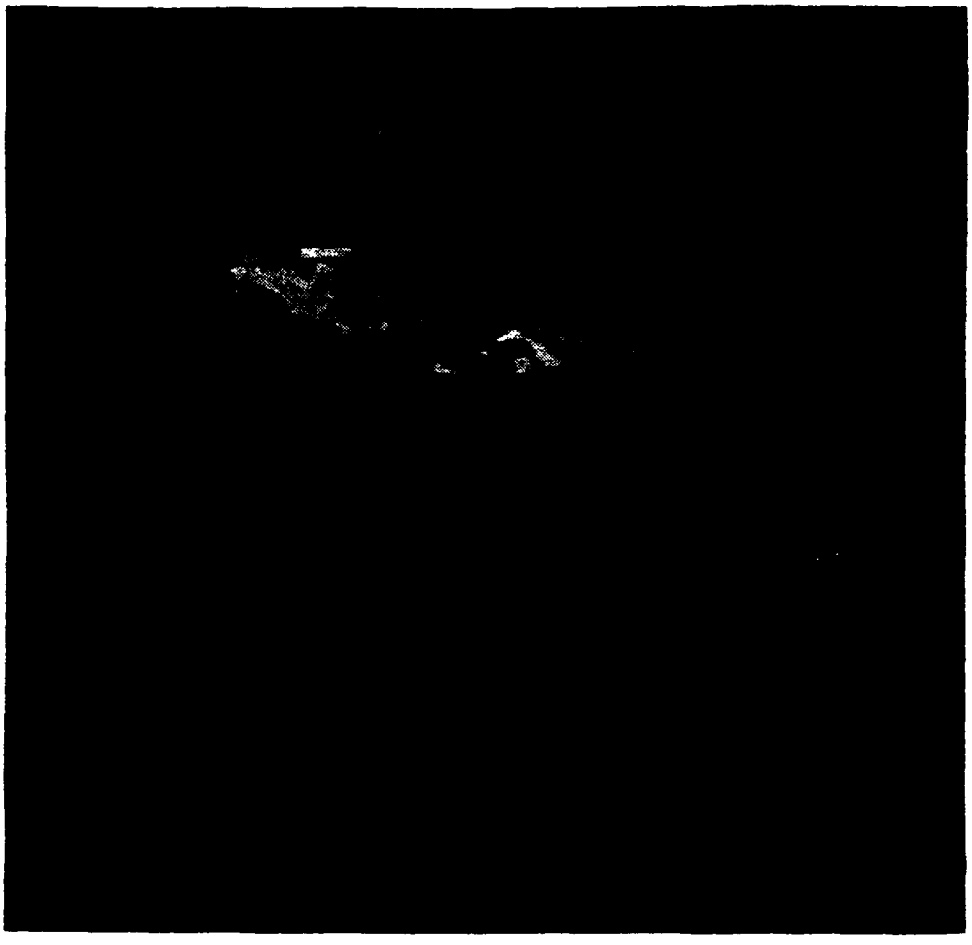

Figure 4.7 Accumulative difference car image.

acceptable for some specific applications, but for low SNR the relative errors are very high and this algorithm does not provide a good velocity estimate.

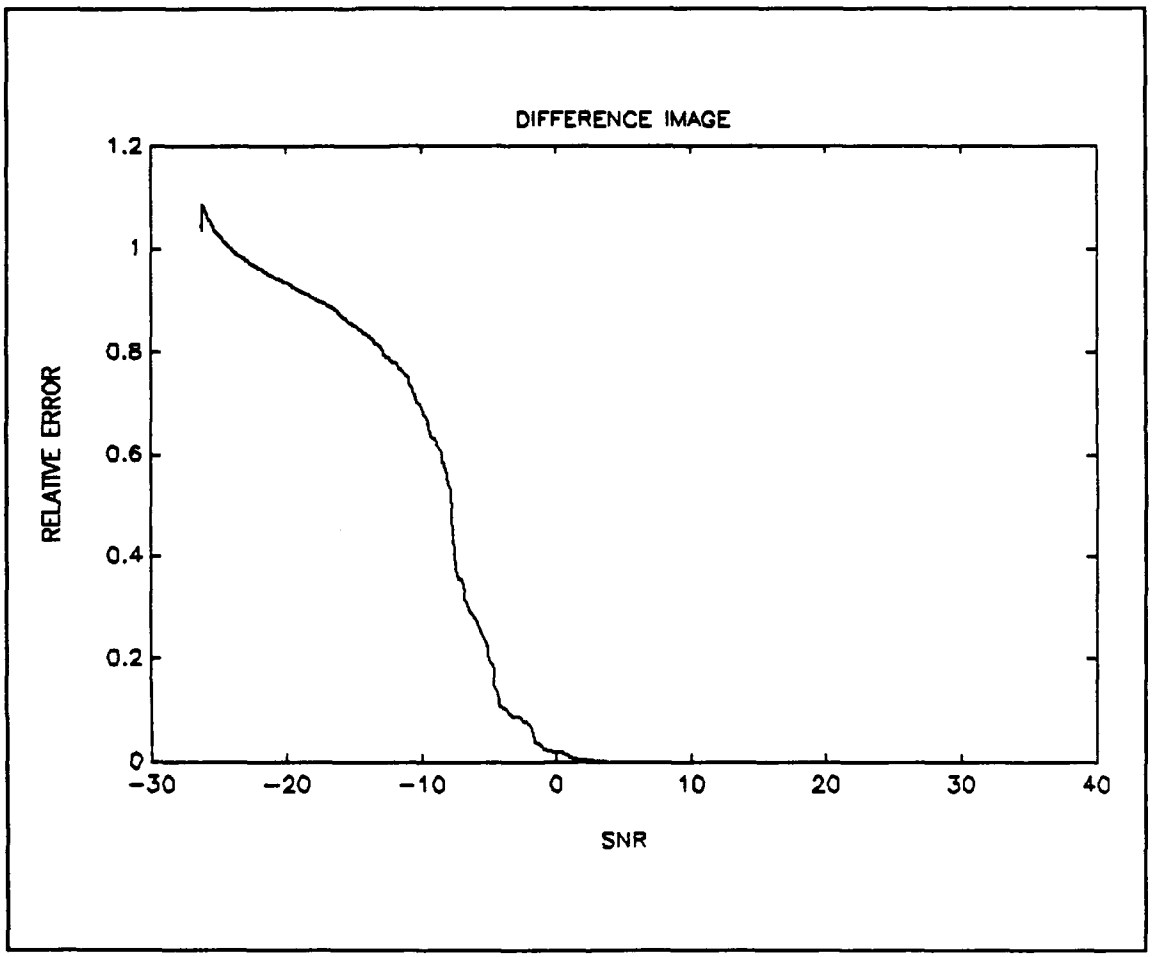

Figure 4.8 Simulation result with threshold 0.50.

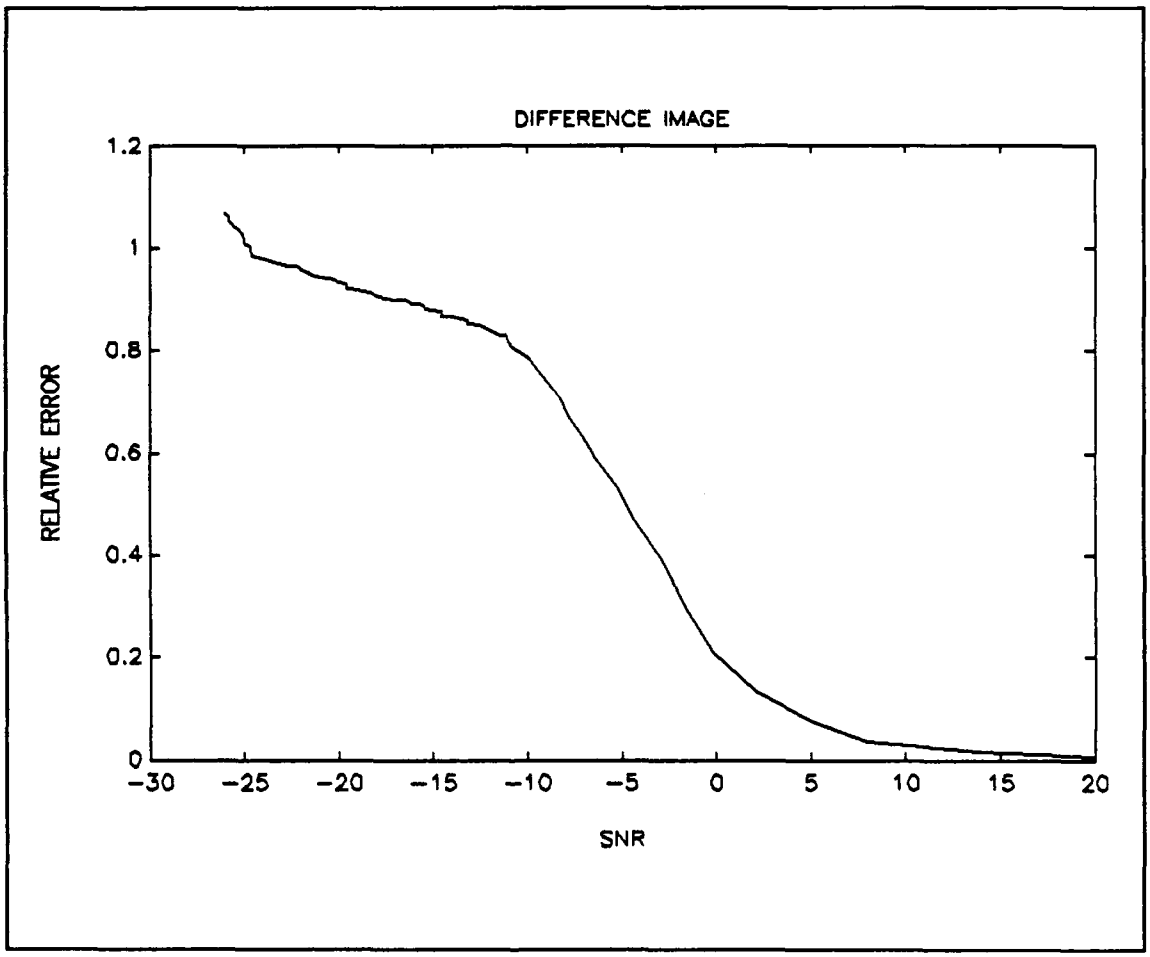

Figure 4.9 Simulation result with threshold 0.25

V. CONCLUSIONS

The performance of the EKF algorithm, linear feature-based algorithms, and the differencing algorithm are evaluated on low SNR images after outlining the algorithms.

Chapter II presents the EKF algorithm which is implemented in the spatial frequency domain. Implementing the EKF in the spatial frequency domain decreases the required computations greatly with respect to a straight forward implementation of the EKF [Ref. 7]. Estimating the object velocity from the frequency-velocity product necessitates the use of more image frames than the other algorithms discussed in this thesis.

The EKF algorithm assumes constant background and provides a two-dimensional nonlocalized velocity vector estimate. Reference 11 summarizes the performance of the algorithm using zero background or a checkerboard background, at varying noise levels. It is shown that as the noise level increases and/or the object in the image plane does not have constant image brightness, such as a pyramid object, the number of iterations required to converge to the true velocity increases greatly.

For low SNR images, the EKF increases the image quality greatly, as shown in the simulation part of Chapter II. This technique produces much better velocity estimates for low SNR images than the other algorithms.

Although a detailed overview of the algorithms based on optical flow are not presented in this thesis, some of the results of Reference 6 will be outlined for comparison purposes. Optical flow algorithms restrict the motion to be smooth and small thus requiring a high rate of image acquisition. It also requires that motion vary

continuously over the image plane. These two restrictions are effected by object occlusion and initial and boundary conditions. Since this algorithm mainly relies on first and second order derivatives of image brightness values, the noise is enhanced during these operations which results in more sensitivity to noise. This algorithm does not provide usable results for low SNR images.

Feature-based approaches, as discussed in Chapter III, strictly require that correspondence be established between image frames. It is the author's belief that much work should be done in this area to improve the performance of these algorithms. It is shown in Chapter III that even small relative perturbations of feature points can result in large relative errors. Noise and occlusion worsen the establishment of correspondences between features and decrease the performance of the algorithms. One way of decreasing the sensitivity to noise is to use more than the required number of features in least squares technique. This can have a smoothing effect but it may also cause additional complications. For example, the computation time is increased for the establishment of correspondences. For ideal cases these algorithms produce very desirable results. These are the only algorithms compared in this thesis that provide 3 dimensional velocity and structure estimation. This characteristic is the main advantage of these algorithms.

The accumulative differencing algorithm provides computationally straightforward motion estimation. Unfortunately only changes in the image scene and the direction of the motion can be estimated with real images. Differencing the images iteratively reduces the SNR of images processed by the algorithm which results in poor estimation.

Using the changes in the COA of binary images also provides a straightforward motion analysis method, but the thresholding and the histrogramming techniques used by the algorithm are vulnerable to noise. In Chapter IV, it is shown that those algorithms do not provide good motion estimation for low SNR images.

The preference of the motion estimation algorithm generally depends on the kind of application and the expected SNR of images. For high SNR images, feature-based algorithms are preferable; they provide three-dimensional information on motion and structure. The perspective projection method can provide realistic results which are similar to those obtained with the human visual system. If the object distance is large, then the variations in the distance to the object points can be ignored and orthographic projection is a good approximation to perspective projection.

Some specific applications such as "early warning systems" may require only the detection of changes in the image plane or low level motion estimation. Then, because of its computational simplicity, differencing algorithms may be preferable. But, for low SNR images, simulations have shown that the EKF provides the best motion and structure estimation. The EKF also enhances the image quality at low SNR. So, for low SNR images, the EKF algorithm is preferable. For future work, the EKF algorithm can be improved to also estimate rotational motions and the depth of the object.

LIST OF REFERENCES

1. Aggaarwal J.K., Nandhakumar N., "On the Computation of Motion from Sequences of Images," *Proceedings of the IEEE,* vol. 293, no. 8, August 1988.

2. Weng J., Huang T.S., Ahuja N., "Motion and Structure from two Perspective Views," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 11, no. 5, 1989.

3. Huang T.S., Lee C.H., "Motion and Structure from Orthographic Projections," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 11, no. 5, 1989.

4. Jacobson L., Wechsler H., "Derivation of Optical Flow Using a Spatiotemporal-Frequency Approach," *Computer Vision, Graphics, and Image Processing*, vol. 83, pp. 29-65, 1987.

5. Klaus B., Horn P., *Robot Vision,* The M.I.T. Press, 1986.

6. Aksu I., Ildiz F., Burl J.B., "A Comparison of Performance of Image Motion Analysis Algorithms Operating on Low Signal to Noise Ratio Images," *paper presented at 34th Midwest Symposium on Circuits and Systems*, Monterey, California, May, 1991.

7. Burl J.B., *A Reduced Order Extended Kalman Filter for Moving Images,* unpublished manuscript, Naval Postgraduate School, Monterey, California, 1990.

8. Gelb A., *Applied Optimal Estimation,* The M.I.T. Press, 1986.

9. Lewis F.L., *Optimal Estimation,* Wiley, 1986.

10. Scharf L.L., *Statistical Signal Processing,* Addison-Wesley, 1991.

11. Lindeman P.A., *A Reduced Order Extended Kalman Filter For Moving Images,* Master's Thesis, Naval Postgraduate School, Monterey, California, 1989.

12. Foley J.D., Dam A., Feiner S.K., Hughes J.F., *Computer Graphics Principles and Practice,* Addison-Wesley, 1990.

13. Rogers D.F., Adams J.A., *Mathematical Elements for Computer Graphics,* McGraw Hill, 1979.

14. Tsai R.Y., Huang T.S., "Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects with Curved Surfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. PAMI-6, no. 1, January, 1984.

15. Longuet-Higgins H.C.,"A computer Algorithm for reconstructing a scene from two projections," *Nature*, vol. 293, pp. 133-135, 1981.

16. Schalkoff R.J., *Digital Image Processing and Computer Vision,* Wiley, 1989.

17. Jain R., Nagel H.H., "On the Analysis of Accumulative Difference Pictures from Image Sequences of Real World Scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. PAMI-1, no. 2, pp. 206-214, 1979.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2. Library, Code 52 Naval Postgraduate School Monterey, California 93943-5002	2
3. Chairman, Code EC Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943-5000	1
4. Profesor J. B. Burl, Code EC/Bl Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943-5000	2
5. Profesor R. Cristi, Code EC/Cx Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943-5000	1
6. Deniz Harp Okulu K.ligi Kutuphanesi Tuzla, Istanbul/TURKEY	2
7. LTJG I. Aksu, TK. Navy K. Harp Okulu Lokali G. M. K. Bulvari No: 48 Maltepe, Ankara/TURKEY	3
8. Doc.Dr. A. H. Kayran Elektrik-Elektronik Fakultesi Maslak, Istanbul/TURKEY 80626	1